NASA TM- 84276

NASA Technical Memorandum 84276

NASA-TM-84276 19830001778

An Integrated User-Oriented Laboratory for Verification of Digital Flight Control Systems — Features and Capabilities

P. de Feo, D. Doane, and J. Saito

August 1982

NASA
National Aeronautics and
Space Administration

NASA Technical Memorandum 84276

An Integrated User-Oriented Laboratory for Verification of Digital Flight Control Systems — Features and Capabilities

P. de Feo
D. Doane
J. Saito, Ames Research Center, Moffett Field, California

National Aeronautics and
Space Administration

Ames Research Center
Moffett Field, California 94035

N83-10048#

TABLE OF CONTENTS

AN INTEGRATED USER-ORIENTED LABORATORY FOR VERIFICATION OF DIGITAL FLIGHT CONTROL SYSTEMS — FEATURES AND CAPABILITIES

P. de Feo, D. Doane, and J. Saito

Ames Research Center

1. INTRODUCTION

This report documents the capabilities of the Digital Flight Control Systems Verification Laboratory (DFCSVL) (fig. 1) which have been assembled at Ames Research Center. The DFCSVL will support research activities in the broad, multidisciplinary area of verification and validation of digital flight control system (DFCS) with the capability to address system and software related subjects.

The major elements of the DFCSVL are:

- A pallet which includes a redundant DFCS, sensor and actuator models, and extensive hardware to support a wide variety of research activities.
- A PDP 11/60 processor digitally connected to the pallet.
- A remote UNIVAC 1100, accessible from the PDP 11/60 through a modem link.
- A chair with pilot controls and limited instrumentation.

The pallet includes extensive capabilities to insert faults in the DFCS and in the sensor and actuator models to support research activities in the area of system verification. A real-time simulation of a wide body, modern transport aircraft, hosted in the PDP 11/60, provides the capability to conduct performance analysis of the DFCS; the pilot chair supports a limited pilot in the loop analysis.

The DFCSVL supports research activities in software verification which include the analysis of automated and semiautomated software verification tools. For this purpose an integrated set of verification tools, consistent with the DFCS software language, has been developed and hosted in the UNIVAC 1100; the compiler, assembler, and link editor for the flight programs are also hosted in the UNIVAC 1100.

The DFCSVL has been designed to simplify as much as possible the user interface to all the resources; the user has been assumed to be a control engineer with no formal training in computer science.

This report is intended to give the reader a good understanding of the research activities that the DFCSVL can support and of the operating scenarios within which these activities can be carried out; the report is not intended to be a user guide nor to be a detailed description of the laboratory.

Figure 1.- Digital Flight Control System Verification Laboratory.

2. PALLET

The pallet is shown in figure 2a; the individual components of the pallet are identified in figure 2b.

This section describes the major components of the pallet in the following order:

Digital flight control system
CAPS test adapter
Modular digital interface control unit (MDICU)
Servo simulator
Glareshield panel
Discrete switch panel
Breakout panel
Buffer panel
Other flight instruments
PDP 11/04

2.1 DIGITAL FLIGHT CONTROL SYSTEM

The digital flight control system is a Collins preproduction unit of the FCS-240 model; it is an integrated system that provides autopilot and flight director modes of operation for automatic and manual control of the airplane during all phases of flight. The FCS-240 includes two identical flight control computers designated as FCC-201; each FCC-201 (fig. 3) includes two CAPS-6 processors, referred to as Channels "A" and "B" (CAPS is the acronym for "Collins Adaptive Processing System").

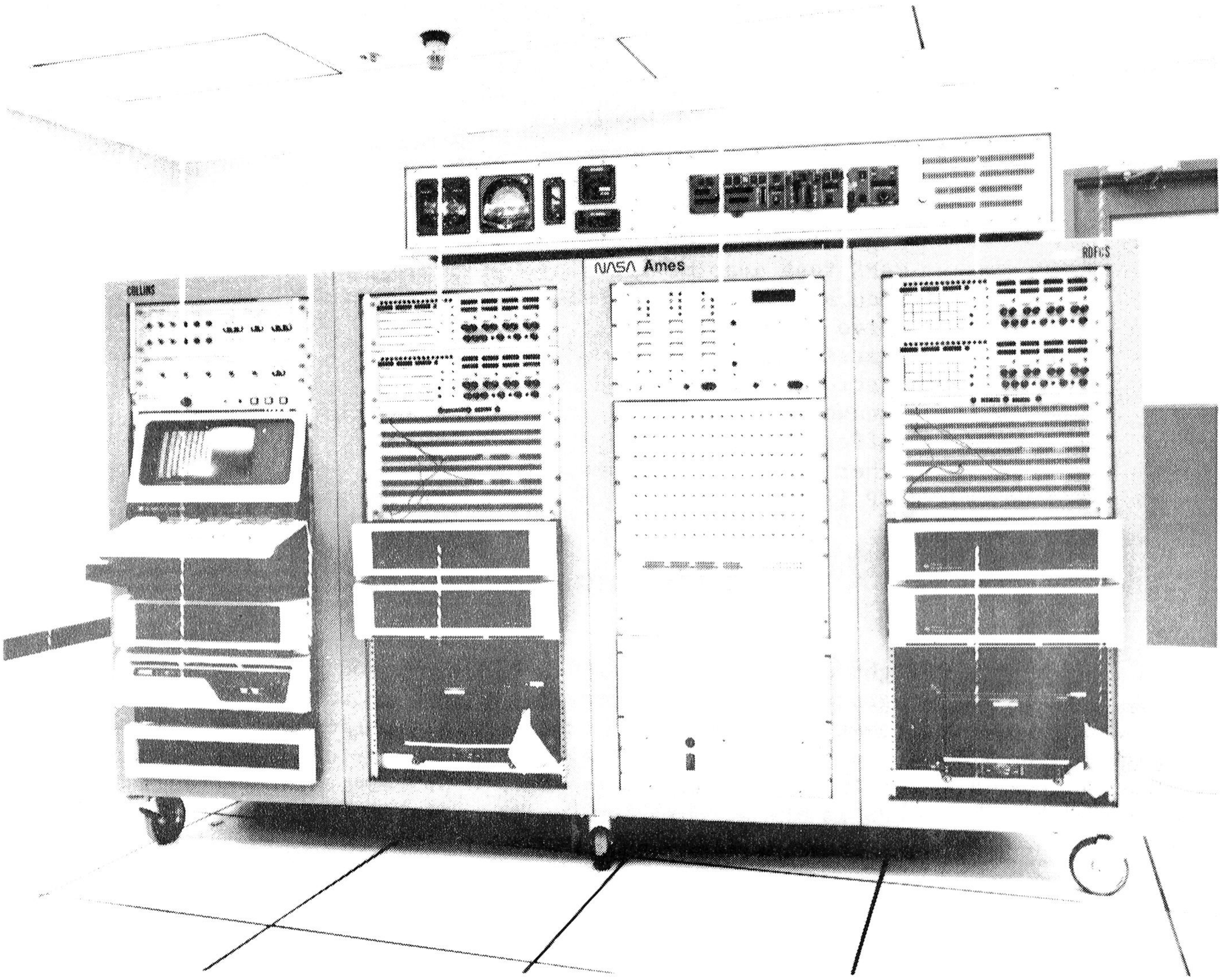
2.1.1 Architecture

The architecture of a flight control system refers to the way the various system components and interfaces are interconnected to accomplish the specified control task. Since no component of any system is immune to failure, accomplishment of the task requires that the system limit the effects of, detect, and in some cases survive, a failure of any system component. The degree to which components must be duplicated depends on the level of criticality of the task and the required survivability of the function. Table 1 defines the various levels of survivability that the FCS-240 system is required to meet.

The system architecture is designed to meet the survivability requirements by providing redundancy of the sensors, actuating systems, interfaces, and processing elements.

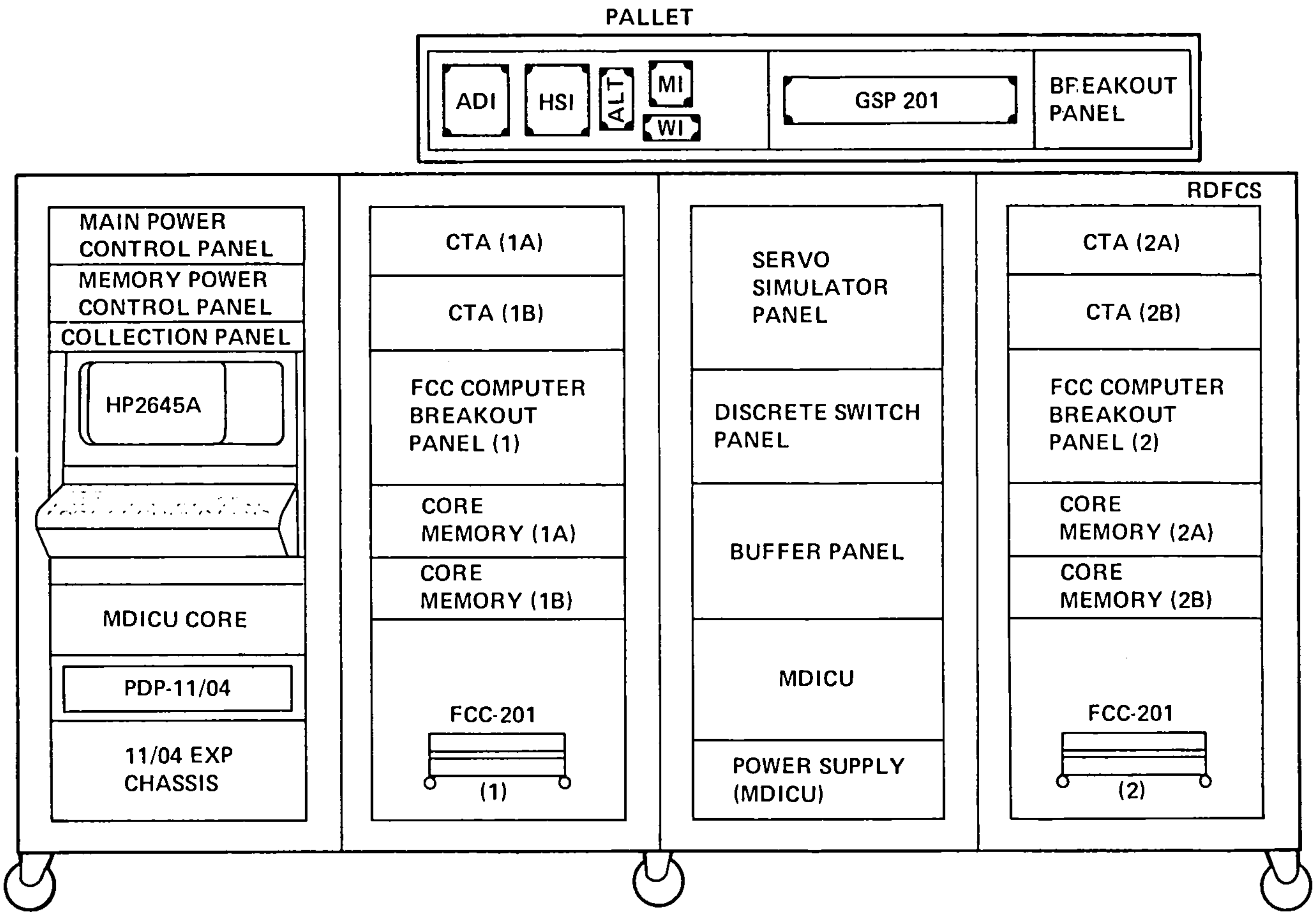
2.1.2 Sensors

The system architecture is configured to handle the following sensor redundancy schemes: quad sensors (four used), triple sensors (three used), and dual monitored sensors (two used).



(a) Pallet.

Figure 2.- Components of the DFCSVL



(b) Pallet components.

Figure 2.- Concluded.

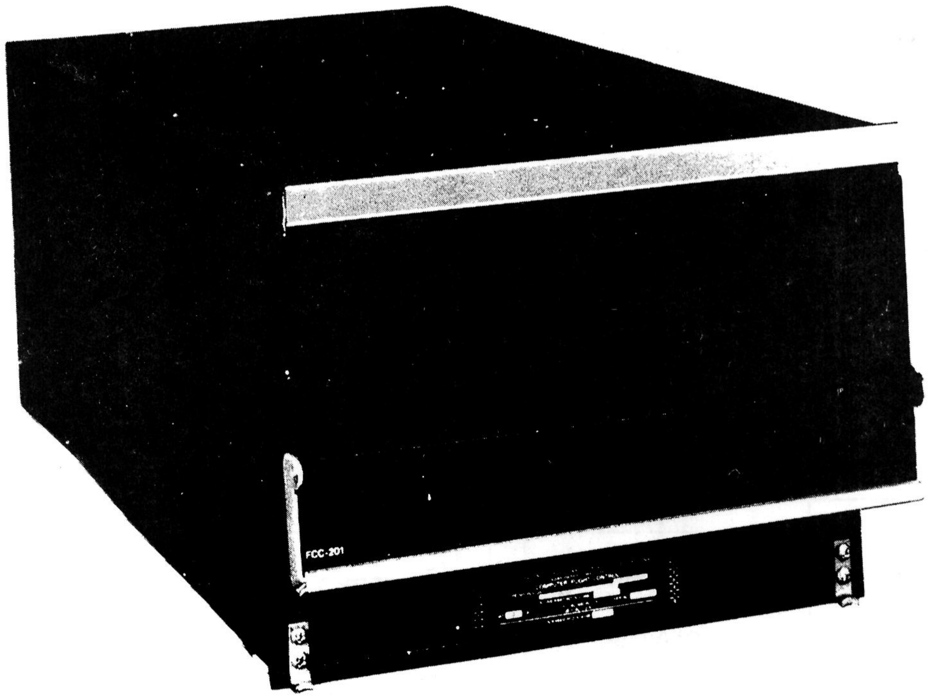


Figure 3.- FCC-201 flight control computer.

TABLE 1.- DEFINITION OF LEVELS OF SURVIVABILITY

Level	Requirement
Fail operational	System functions normally after any single component or interface fails. After the first failure, a fail operational system becomes fail passive.
Fail passive	System will not perturb the aircraft as a result of any single failure. The system may, however, disconnect itself, leaving the aircraft in a trimmed condition, so that essentially no change in aircraft motion or control surface occurs.
Fail soft	System will withstand any failure without endangering passenger safety, or causing a dangerous deviation from the flightpath if reasonable pilot attention is provided.

Figure 4 shows how the different sensor schemes are handled in the dual-dual processor configuration of the FCS-240. The usual way of detecting sensor failures is by comparing the outputs of two (or more) sensors measuring the same parameter. If a difference is detected, a sensor has failed. If three or more sensors are used, the failed sensor may be determined by multiple comparisons. Thus, since three sensors are the minimum required to achieve a fail-operational status, that is the number generally used.

A dual monitored sensor provides two buffered outputs to a single, in-line monitored, internal "node," or tie point. The device itself contains monitoring of sufficient integrity that no single fault can cause the sensor to generate an erroneous parameter value at its internal node without also causing the monitoring to detect the fault and indicate the faulted condition to the using equipment. In a dual-dual processor configuration two dual monitored sensors are provided, one for each FCC.

Quad sensors are used in lieu of triple sensors where other system constraints preclude the use of triple sensors. Triple sensors require three independent sources of power and total independence of the measuring system. If three independent sources of power are not available or if, as in the case of servo position sensors, a physical position of a linkage is being measured, quad sensors may be used. Table 2 lists the redundancy levels of the various sensors associated with the FCS-240.

The input processing for quad and triplex configurations consists of:

1. A series of logic switches to disconnect faulted signals and reconfigure the remaining valid sensors (fig. 5a). The switching logic is a function of (1) the sensor configuration (triplex, dual-dual, dual-self-monitored), (2) the output of the signal comparators, and (3) the status of sensor validity flags.

2. Voting algorithms for signal selection (fig. 5b). The voters contain two sections: the first section selects the most positive signals from the four input signals: 1, 2, 3, and 4; the second section selects the most negative output signal from the first section. Note that the chosen output is always one of the two midvalues with the preference being given to inputs 1 and 4 over 2 and 3 because, for a triple sensor configuration, the inputs numbered 2 and 3 are both originated from the same sensor. Also shown in figure 5b are the selected signals for every possible combination of relative values of the signal sources.

3. Equalizing algorithms. These algorithms are used to avoid degraded system performance after a first sensor failure and to remove sensor bias effects. These algorithms slowly change the output of each sensor to approximate the sensor computed midvalue by adding to each sensor output an equalization signal which is the limited integral of the difference between the computed midvalue and the sensor output.

4. Voting algorithms. These new sets of voters act on the equalized signal values in the same way that the previous voters act on the unequalized signals. The processing of dual sensors does not include the equalizing algorithms.

2.1.3 Actuating System

The equalization is also performed on the computed output commands to reduce the effects of computational differences between channel. Output equalization is provided in pitch and roll axes during autoland mode of operation only. The output

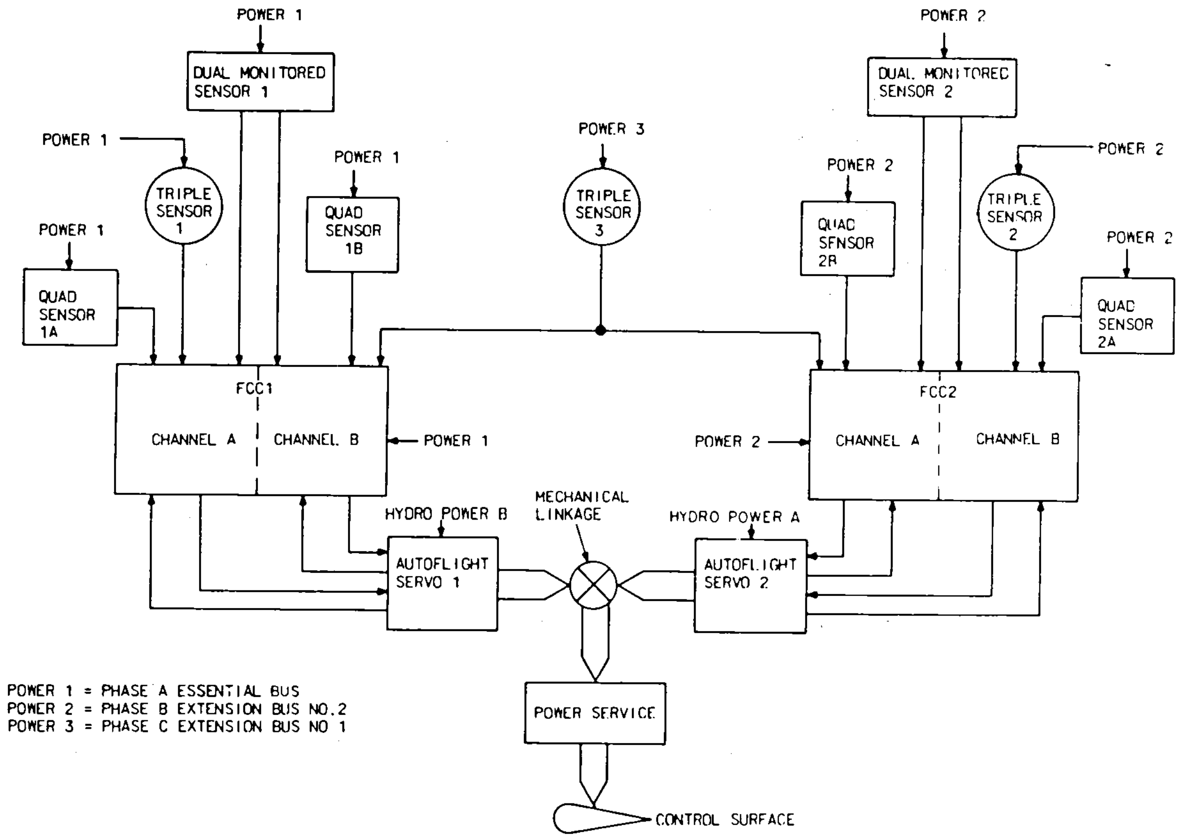
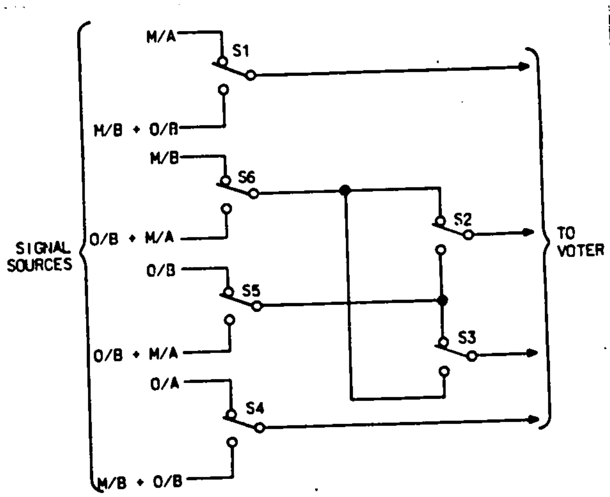


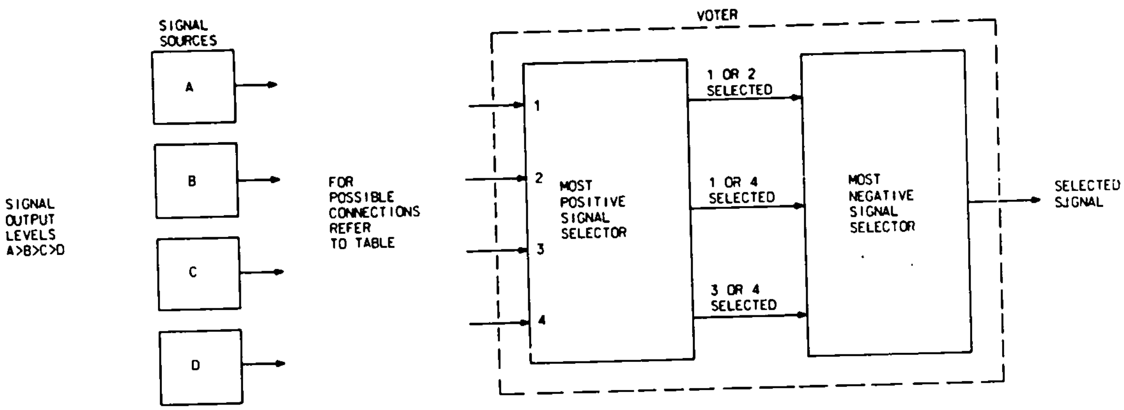
Figure 4.- Dual-dual architecture.

TABLE 2.- SENSOR REDUNDANCY LEVEL

Sensor	Redundancy level
Instrument landing system (ILS)	Dual monitored
Radio altimeter	Dual monitored
Pitch, roll, yaw servo and amplifier	Dual monitored
Attitude (pitch, roll)	Triple
Acceleration (lateral, normal)	Triple
Yaw rate	Triple
Surface position (rudder, aileron)	Quad (two dual sensors)
Trim error (column minus trim)	Quad (two dual sensors)



(a) Voter switching.



		POSSIBLE CONNECTIONS																						
INPUTS TO MOST POSITIVE SIGNAL SELECTOR	1	A	A	A	A	A	A	B	B	B	B	B	B	C	C	C	C	C	C	D	D	D	D	D	D
	2	B	B	C	C	D	D	A	A	C	C	D	D	A	A	B	B	D	D	A	A	B	B	C	C
	3	C	D	B	D	B	C	C	D	A	D	A	C	B	D	A	D	A	B	B	C	A	C	A	3
	4	D	C	D	B	C	B	D	C	D	A	C	A	D	B	D	A	B	A	C	B	C	A	B	A
OUTPUT FROM MOST POSITIVE SIGNAL SELECTOR	1 OR 2 SELECTED	1/A	1/A	1/A	1/A	1/A	1/A	2/A	2/A	1/B	1/B	1/B	1/B	2/A	2/A	2/B	2/B	1/C	1/C	2/A	2/A	2/B	2/B	2/C	2/C
	1 OR 4 SELECTED	1/A	1/A	1/A	1/A	1/A	1/A	1/B	1/B	1/B	4/A	1/B	4/A	1/C	4/B	1/C	4/A	4/B	4/A	4/C	4/B	4/C	4/A	4/D	4/A
	3 OR 4 SELECTED	3/C	4/C	3/B	4/B	3/B	4/B	3/C	4/C	3/A	4/A	3/A	4/A	3/B	4/B	3/A	4/A	3/A	4/A	3/B	4/B	3/A	4/A	3/A	4/A
OUTPUT FROM MOST NEGATIVE SIGNAL SELECTOR		C\3	C\4	B\3	B\4	B\3	B\4	C\3	C\4	B\1	B\1	B\1	B\1	C\1	B\4	C\1	B\2	C\1	C\1	C\4	B\4	C\4	B\2	C\2	C\2

(b) Voter logic.

Figure 5.- Sensors.

equalization scheme is identical to the one used for sensor equalization (fig. 6). The output signals from the two channels (A and B) within the same FCC are averaged by the hardware prior to being utilized as input to the servo loop.

The integrity of the hardware is tested by monitoring: (1) the modulation piston-rate commands against the electrohydraulic coil current to rapidly detect jams, runaways, and loss of hydraulic power in the servos and (2) the servo amplifier coil currents to detect the electrical integrity of the circuits (fig. 7).

2.1.4 Interfaces

Communications between the two channels within each FCC and between the two FCCs are performed via dedicated ARINC 429 type buses (16 data bits, 8 address bits). These buses are asynchronous, unidirectional, and nonredundant.

2.1.5 Processing Elements

The processor of the flight computer is a CAPS-6 model of a medium-speed processor employing bit slice large scale integration (LSI) components. The CAPS-6 is a stack oriented, 16-bit, microprogrammed machine with the following general features: 250 nsec microcycle, 1024×40 control store for microprogramming, 17 general purpose registers, 8 priority interrupts with mask capabilities, 93 standard instructions, 64K words addressable space, and 289 KOPS (15% multiply/divide and 15% double precision is assumed).

The primary feature of the CAPS-6 processor is its heavy stack orientation. The process variables must be pushed into stacks before they are manipulated by the central processing unit (CPU). Most of this process is invisible to a programmer operating in a high-order language environment; however, debugging and patching programs are more cumbersome in the CAPS-6 than in conventional processors. The major advantages of the stack architecture are a highly efficient compiler code generation and a wide, variable accessibility. The interfaces among the CAPS-6 processors, memories, and peripheral devices are performed through the CAPS-6 transfer bus (fig. 8). The communication protocol is the same for every device on the bus, each employing a standard interface to common address lines (16), data line (16), and control paths (9). Two classes of devices connect to the transfer bus — master devices that control the data transfers and slave devices that supply or accept data in response to a master's request. Data transfers in either direction always occur between one master and one slave; at any point in time only one device (master) can have control of the bus and communication flows between that master and the slave selected by the master. The CAPS-6 CPU is a master device, whereas a memory module is a slave device. The masters are dynamically activated one at a time as their need for a data transfer arises; slave devices are assigned one or more addresses on the bus and remain passive until specifically addressed by a master.

2.1.6 Flight Software

All the flight software is written in AED (automated engineer design), an Algol derivative high-order language; the only exception is the CPU diagnostic program, a background program written in assembly language. The flight programs in channels "A" and "B" require 16,500 and 15,000 words of storage, respectively.

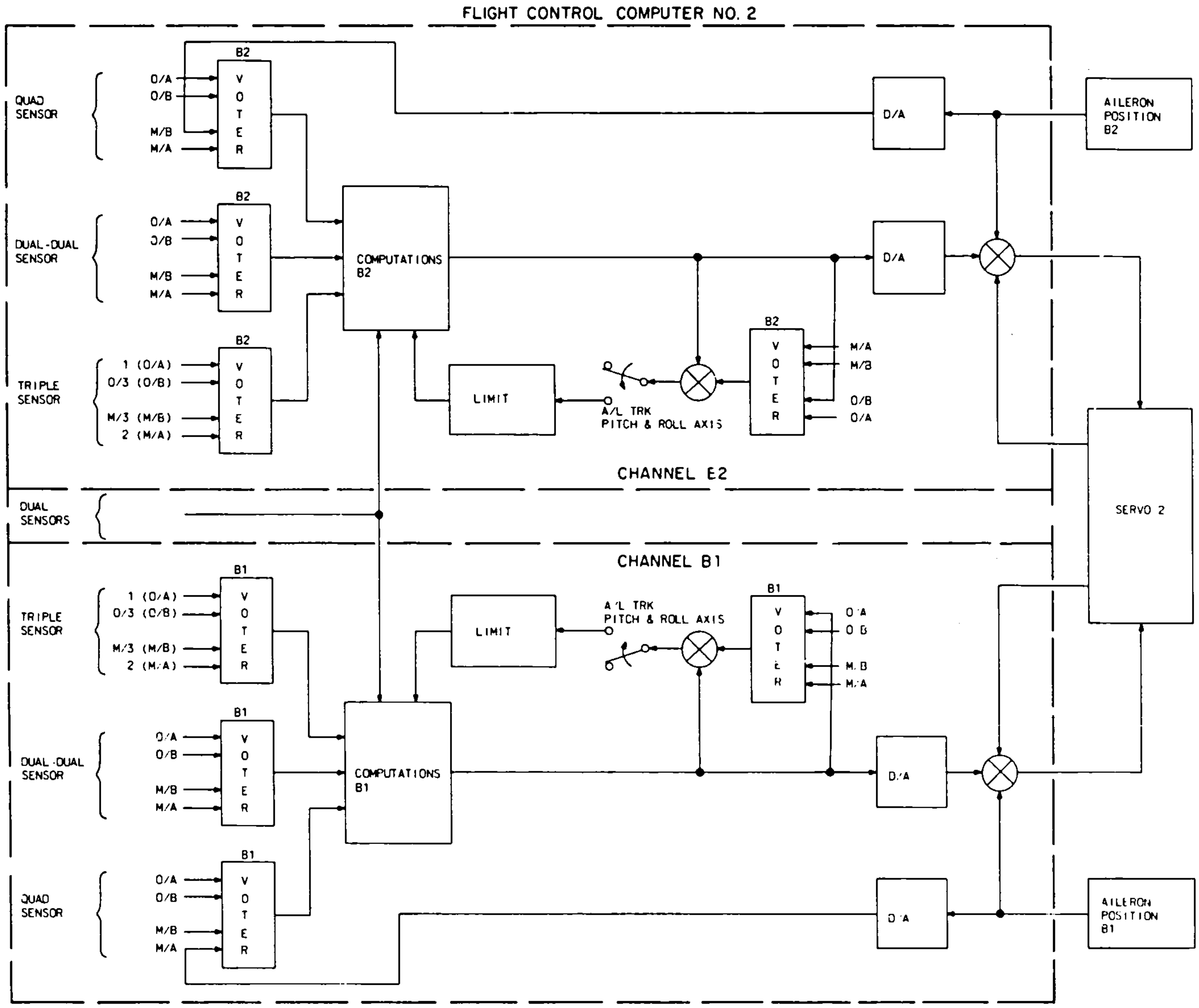


Figure 6.- Sensor and output equalization scheme.

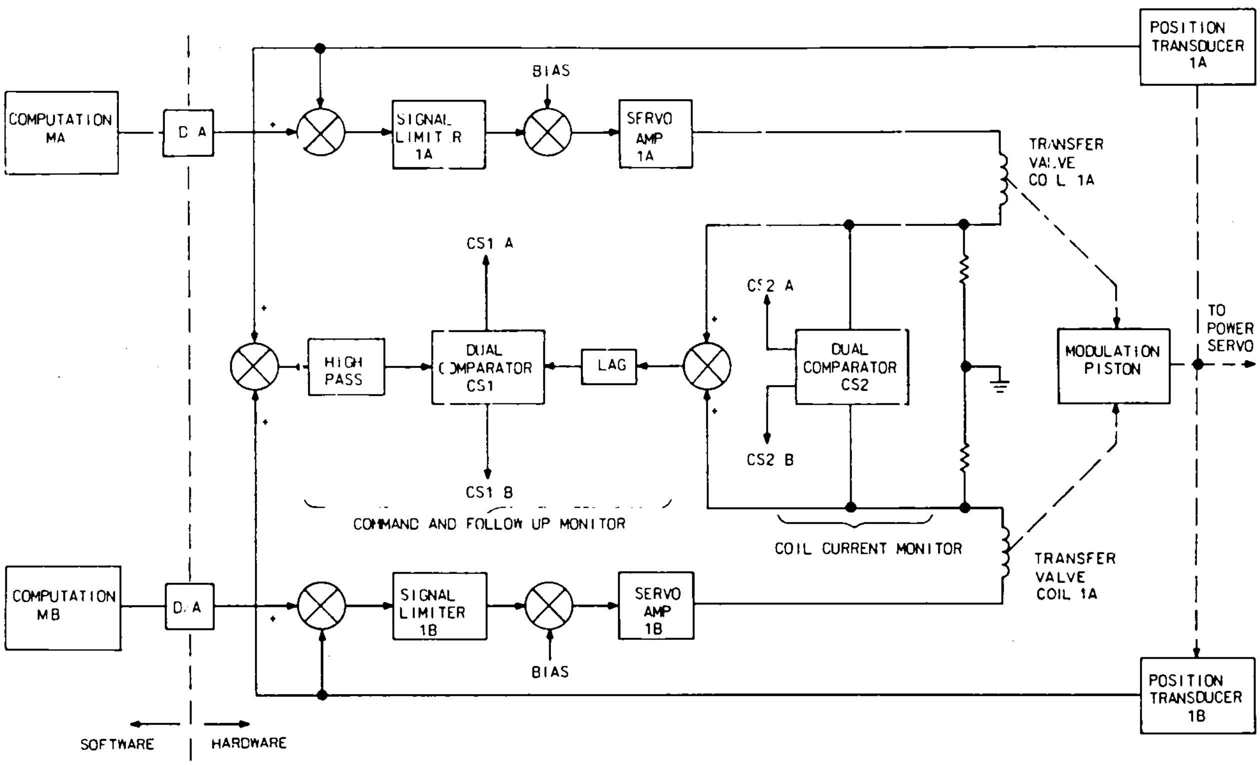


Figure 7.- Servo loop monitor.

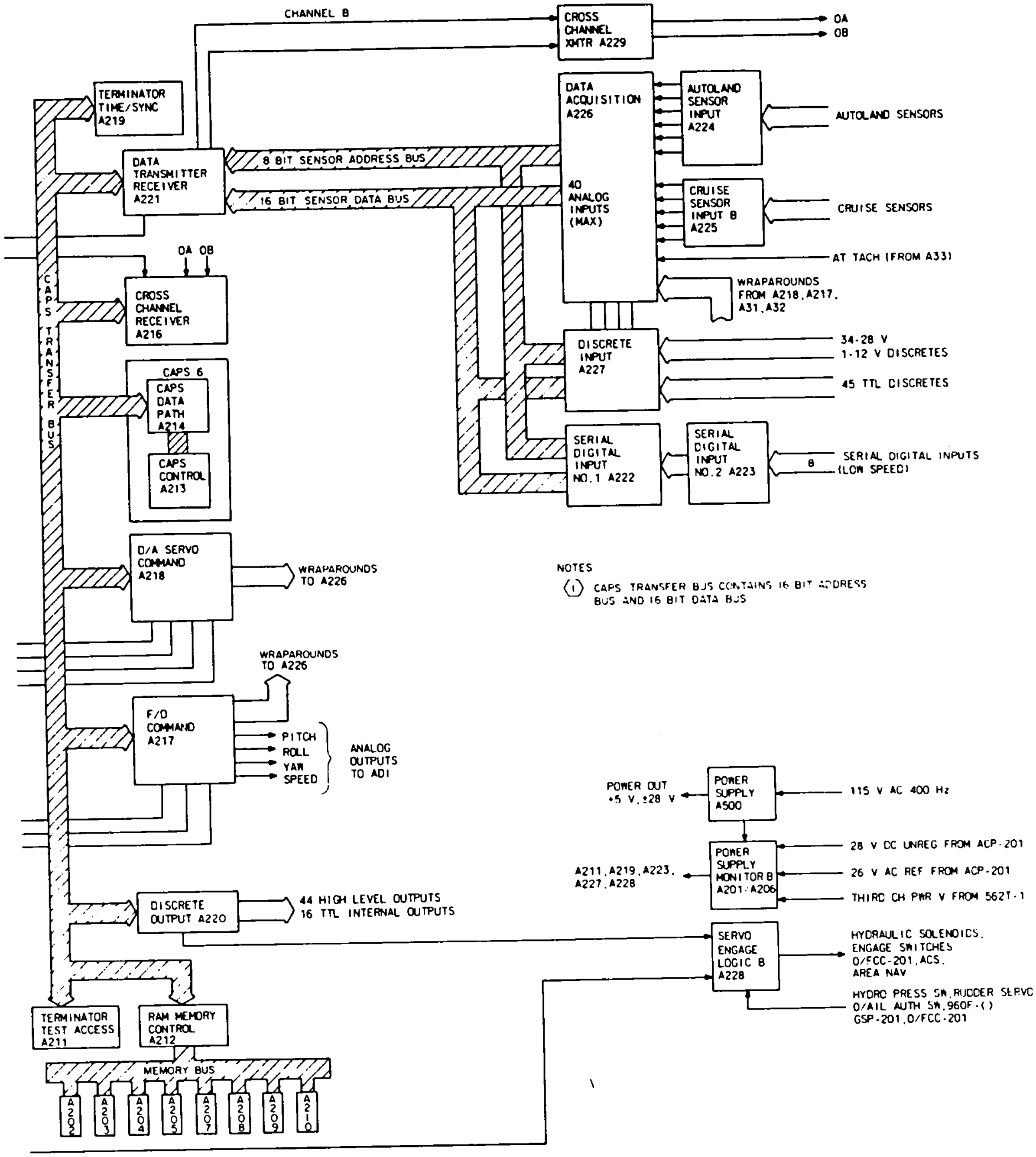


Figure 8.- CAPS transfer bus.

Table 3 shows in which channel each major program is implemented; all the critical programs are implemented in both channels. Table 4 shows the partitioning of the flight software, from the functional point of view, into five major categories:

1. Control and navigation— Modules in this category perform primarily or entirely computations for the aircraft automatic control and navigation, variable filtering, gain schedules, algorithms, etc.

2. Logic— Modules in this category perform exclusively engage- and mode-logic computations and use Boolean statements only.

3. Testing and voting— Modules perform real-time tests on CPU, memory, sensors, and actuators; they manage and control the system configuration as a function of outstanding detected failures.

4. I/O— Modules perform data handling and formatting, data transmission and display.

5. Executives— Modules perform a multiplicity of executive tasks such as initialization procedures, system tests at power-up, synchronization, timing, and scheduling. The operating system is included in this category.

The memory requirements for each of the five categories and for variables and stacks allocations are also shown in table 4. The software from an executive point of view is organized in five major categories — the operating system and the executive, the foreground application programs, the sixty program, the error service program, and the background application programs.

2.1.6.1 The Operating System and the Executive

The operating system and the executive control the scheduling and the execution of all the other programs. The operating system is basically an interrupt handler that controls which program runs at any given time (foreground, background, sixty, or error programs), based on time interrupts, halt interrupts, and error interrupts (fig. 9). The executive controls the execution of the programs included within the foreground application based on the fixed-path organization outlined in figure 10.

The two flight control computers (FCC-201) which comprise the flight control system (FCS-240) are not synchronized; however, three algorithms are implemented to synchronize the two channels within each flight control computer. These are:

1. The time-synchronization algorithm that slaves channel "A" to start every 52 msec or at channel "B" restart (frame synchronization).

2. The data-synchronization algorithm that ensures that inner-loop computations in either channel receive consistent outer-loop commands. This is necessary because during cruise the two channels, "A" and "B," process different outer-loop algorithms; however, the inner-loop computations, duplicated in both channels, need the outputs of both channels. This synchronization is achieved by having one processor waiting, prior to starting the inner-loop computations, until the other processor is also at the wait point or when a maximum of 52 msec is exceeded.

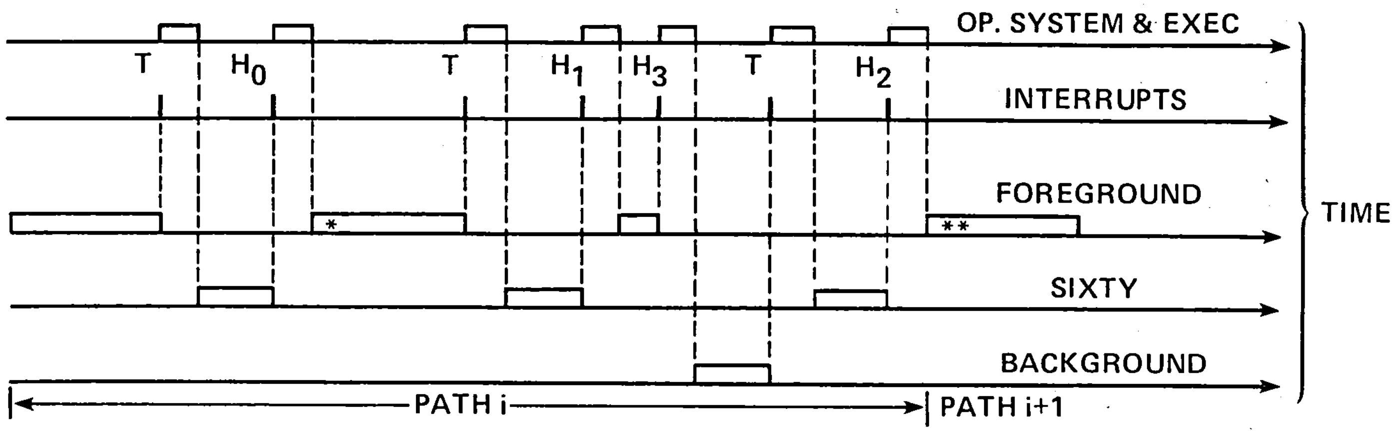
TABLE 3.- FLIGHT SOFTWARE FUNCTIONS

Function	Channel "A"	Channel "B"
Pitch autoland (CAT III)	X	X
Roll autoland (CAT III)	X	X
Yaw autoland (CAT III)	X	X
Takeoff and go-around (TOGA)	X	X
Engage logic	X	X
Servo monitoring	X	X
Pre-engage test[a]	X	X
Self test[a]	X	X
Synchronization	X	X
Instrumentation	X	X
Annunciation	X	X
Yaw SAS	X	X
Inner loops	X	X
Fault isolation[a]	X	X
Pitch cruise outer loop		X
Autothrottle		X
Roll cruise outer loop	X	
Alt alert	X	
Mode logic	X	
Glareshield interface	X	
Maintenance computer driver[a]	X	
Nonexecutive sensor comparisons	X	

[a]Not available in the DFCSVL software.

TABLE 4.- FUNCTIONAL PARTITIONING OF FLIGHT SOFTWARE

Function	Channel "A"		Channel "B"		Channels "A" and "B"	
Control and navigation	4384	26%	4767	30%	9151	27%
Logic	4365	26%	1333	9%	5698	18%
Testing and voting	2498	15%	4480	28%	6978	22%
I/O	1762	11%	348	2%	2710	7%
Executives	1534	9%	2072	13%	3606	11%
Stack and variables	2105	13%	2801	18%	4906	15%



LEGEND:

* IT INCLUDES WAIT FOR DATA SYNCHRONIZATION

**IT INCLUDES WAIT FOR TIME SYNCHRONIZATION

H_0, H_1 = OPERATING SYSTEM RESCHEDULES THE INTERRUPTED TASK

H_2 = OPERATING SYSTEM RESCHEDULES THE FOREGROUND (END OF FRAME)

H_3 = OPERATING SYSTEM RESCHEDULES THE BACKGROUND (FOREGROUND COMPLETED)

T = TIME INTERRUPT FOR SCHEDULING THE SIXTY PROGRAM

Figure 9.- Program schedule.

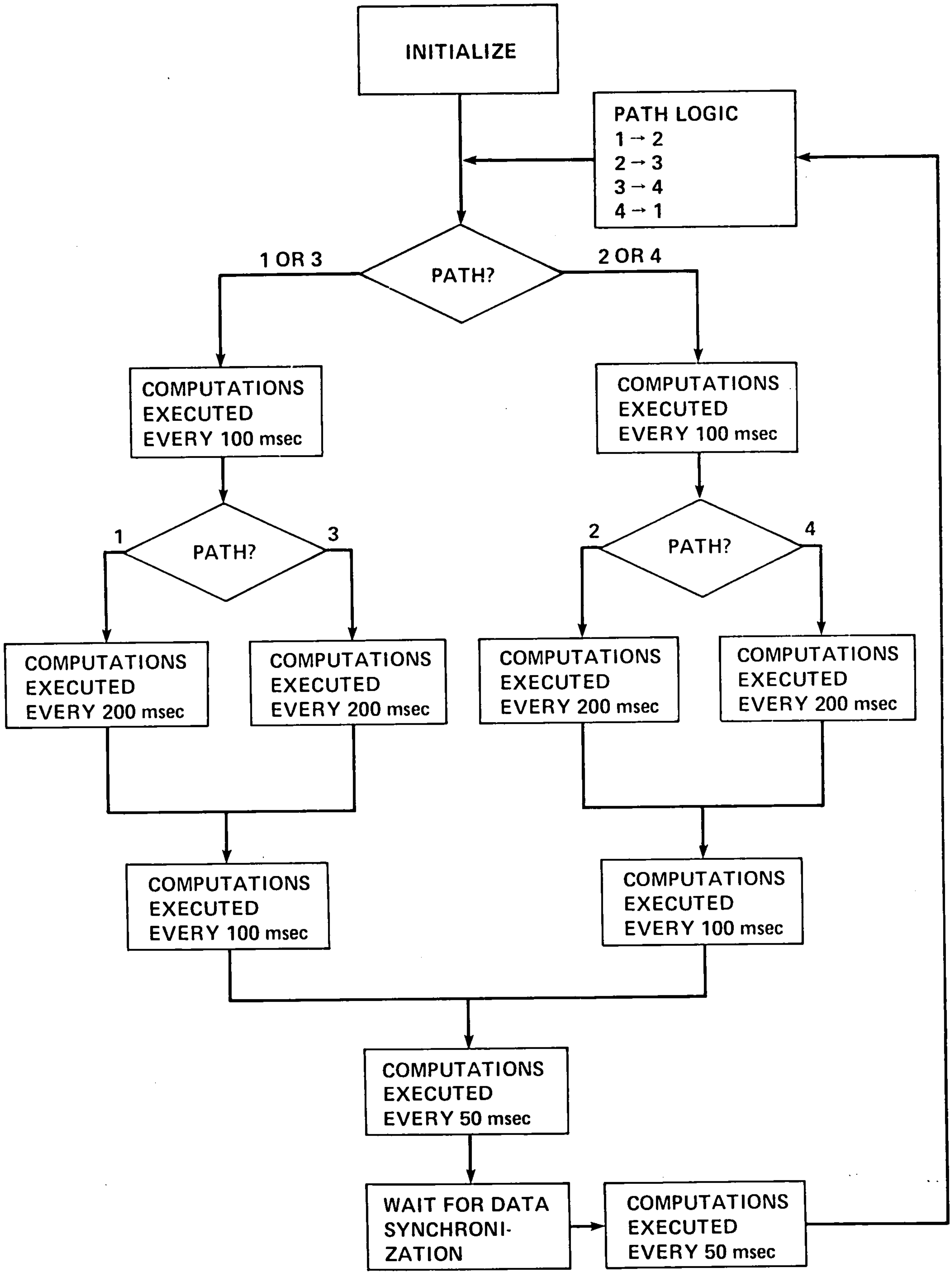


Figure 10.- Executive control.

3. The path synchronization algorithm that ensures that both channels always execute the same path number. To achieve this, channel "B" transmits its path number to channel "A" which latches to it.

2.1.6.2 The Foreground Application Programs

The foreground application programs require 80% and 50% of the total storage available for channels "A" and "B," respectively. The execution time of the foreground varies from 70% to 75% of the main frame time of 50 msec, depending on flight regime and mode selection. All the autopilot functions, except those implemented in the sixty program, are included in the foreground. Foreground programs are executed at three different computation rates depending on the dynamic content of the algorithms — every 50, 100, or 200 msec.

Inner-loop computations, including yaw SAS, synchronization algorithms, and computations supporting common axis modes (approach, land, takeoff, go-around) are executed every 50 msec; engage logic and longitudinal autoland computations are executed every 100 msec; monitoring algorithms, lateral outer loops, annunciation, and mode select computations are executed every 200 msec.

The FCS-240 operates in three different modes — control wheel steering (CWS), command (CMD), and flight director (FD). In the CWS mode the pitch and roll attitudes of the airplane are normally held constant but they can be changed by applying an appropriate force to the control wheel. In the CMD mode of operation the airplane is automatically controlled to computed guidance commands. In the FD mode the attitude direction indicator (ADI) command bars are driven to provide the pilot with visual cues for manual control to a computed flightpath.

The functions included in the autopilot are listed in table 5. An automatic pitch trim mode, which relieves steady loads on the autopilot servos, is provided during the operation of all the pitch axis functions. The autopilot and flight director command mode engagement logic is shown in table 6.

The following is a higher level description of the major functions within the pitch, roll, and common axis modes and within the autothrottle mode.

2.1.6.2.1 Pitch axis modes— The following pitch axis control modes are available:

Pitch attitude hold	Vertical speed
Altitude hold	Indicated airspeed hold
Altitude select	Mach hold

The pitch attitude hold mode is the basic pitch control mode and is operative when either autopilot is engaged in the basic (CWS) configuration or either or both flight directors are engaged with no other pitch mode selected. The other modes listed above are selectable using the mode select pushbuttons on the glareshield panel.

Automatic pitch trim is provided during operation in all the pitch axis modes when either autopilot channel is engaged in the basic or command configuration. The automatic trim system acts to relieve any load on the autopilot servos to prevent transients when the autopilot is either manually or automatically disengaged.

TABLE 5.- AUTOPILOT FUNCTIONS

Pitch axis	Lateral axis	Common axis
Pitch attitude hold	Heading/bank angle hold	Approach/land
Altitude hold	Heading select	Approach
Altitude select	Localizer	Go-around
Vertical speed	Back course	Takeoff
Indicated airspeed hold	VOR	Turbulence
Mach hold	Lateral navigation (LNAV)	
Vertical navigation		

TABLE 6.- AUTOPILOT/FLIGHT DIRECTOR COMMAND MODE ENGAGEMENT FUNCTIONS

Autopilot engagement status	Flight director engagement status	Selected command mode	Result
OFF	OFF	Will not engage	ADI command bars out of view.
	ON	None	ADI command bars out of view.
	ON	Takeoff or go-around	ADI command bar visual cues for manual control to a wings-level roll attitude and an optimum rotation and climb-out maneuver.
	ON	Any other (see notes 1 and 3)	Roll ADI command bar visual cues for manual control to a computed guidance signal.
BASIC (CWS)	OFF	None	Automatic pitch attitude and roll attitude/heading hold with control wheel steering. ADI command bars out of view.
	ON	None	Same as above.
	ON	Takeoff or go-around	Automatic pitch attitude and roll wing level with control wheel steering. ADI command bar visual cues for manual control to a wings-level roll attitude, and an optimum rotation and climb-out maneuver.
	ON	Altitude hold and capture	Automatic altitude capture and hold. Roll attitude-heading hold with control wheel steering. ADI command bars out of view.
	ON	Turbulence	Automatic pitch and roll attitude hold with control wheel steering, both at reduced gain levels. ADI command bars out of view.
	ON	Any other (see note 3)	Automatic pitch attitude and roll attitude/heading hold with control wheel steering. Roll ADI command bar steering. Roll ADI command bar visual cues for manual control to a computed flightpath.
COMMAND (CMD)	OFF	Any other (see note 2)	Automatic control to a computed signal. ADI command bars out of view.
	ON	Any other (see notes 2 and 3)	Automatic control to a computed guidance signal. Roll ADI command bars follow A/P commands.

The pitch attitude hold mode is used to maintain the aircraft pitch attitude existing at the time of engagement. It may be used during all phases of the flight regime (takeoff, climb, cruise, descent, holding pattern, etc.). It is compatible with any roll axis cruise mode.

The altitude hold mode is used to maintain the aircraft barometric altitude existing at the time of mode selection. It is selectable for either autopilot or flight director control and is compatible with any roll axis mode.

The altitude select mode is used to acquire a preselected altitude that can be used for either autopilot or flight director control. This mode, in conjunction with a precapture pitch guidance mode, is most commonly used during the climb and descent phases of the flight regime. It is compatible with any roll axis mode.

The vertical speed mode is used to maintain the vertical speed existing at the time the mode is established. It is selectable for flight director and autopilot control and is most commonly used during the climb and descent phases of the flight regime. It is compatible with any roll axis mode.

The indicated airspeed hold mode maintains the airplane speed existing at the time of mode selection. It is selectable for either autopilot or flight director control and is most commonly used during the low altitude cruise portion of the flight regime. It is compatible with any roll axis mode.

The Mach hold mode maintains the Mach number existing at the time of mode selection and is selectable for either autopilot or flight director control. It is most commonly used during the high-altitude cruise phase of the flight regime and is compatible with any roll axis mode.

2.1.6.2.2 Roll axis modes– The following roll axis modes are available:

Basic mode (heading hold/bank angle hold)
Heading select
Localizer (LOC)
Back course LOC (BCK CRS)
VOR

The basic mode of operation is acquired with either autopilot engage switch in the CWS or CMD position. If in the CMD position, this mode of operation occurs only when no other roll axis command mode has been selected. The computer will hold heading when the roll attitude is less than 3° and when the roll force on the control wheel is less than 1.26 kg (2.8 lb). Bank angle hold mode is engaged in the same manner as heading hold except the roll attitude must be greater than 3°.

The heading select mode allows the aircraft to be flown to any heading selected by operating the dedicated pushbutton located on the glareshield panel. It is engaged after the following conditions are met:

1. Either or both flight directors are engaged and/or either autopilot is engaged in the command mode.

2. No higher priority roll mode is established (turbulence or the approach/land mode after localizer capture).

The localizer mode is used to acquire and track an instrument landing system (ILS) azimuth guidance signal down to Category I minimums or to operationally allow passing through the glide slope for above-beam glide-slope capture.

The back course localizer mode is used where flight director lateral guidance is desired for an approach using the back beam of the localizer.

The VOR mode is used for navigation guidance to acquire and track a VHF omnirange (VOR) radial. VOR deviation is calculated in the digital flight control computers based on VOR heading from the VOR receiver, selected course from the glareshield panel, and the aircraft heading.

2.1.6.2.3 Common axis modes— The go-around mode provides vertical guidance and fast/slow commands to rapidly and safely arrest the aircraft descent rate as well as to produce a climb rate commensurate with the aircraft speed and pitch attitude. Wings level commands are provided for roll axis control in this mode. Selection of the go-around mode automatically releases all other modes except turbulence.

The takeoff mode (a flight director mode only) provides flight director pitch and fast/slow displays for an optimum takeoff and climbout maneuver with wings level commands for roll axis control. The pitch and speed control laws and reference displays are the same used for go-around mode.

The turbulence mode is an autopilot mode only and is normally used when the aircraft is flying in turbulence. When this mode is established, the autopilot reverts to the basic configuration with reduced gains to provide softer control. The control wheel steering force levels and dead zones are also increased to 3.6 kg (8 lb) to reduce the potential for overcontrol of the aircraft. Once the turbulence mode is established, no other mode may be engaged until the turbulence mode is manually released.

2.1.6.2.4 Yaw stability augmentation mode— The yaw stability augmentation system (Yaw SAS) provides basic aircraft yaw damping and turn-coordination control, as well as load alleviation for the vertical stabilizer.

In the basic stability augmentation mode of operation, the flight computer provides rudder control to damp out the natural Dutch roll tendencies of the aircraft. Turn coordination is also provided by applying rudder commands to prevent the aircraft slipping into the turn.

2.1.6.2.5 Autothrottle modes— The FCS-240 digital AFCS provides fail-passive control of the automatic throttle system (ATS) in the following modes of operation — indicated airspeed, stall margin, thrust management, and flare.

The autothrottle is engaged by pressing the dedicated autothrottle pushbutton on the glareshield panel. If the required interlocks are satisfied, the autothrottle servo is energized.

The autothrottle indicated airspeed (IAS) hold mode is engaged to control the throttles to maintain the existing airspeed at time of engagement.

The stall margin mode of operation is acquired automatically when the airspeed, when related to the configuration of the flaps and slats on the aircraft, is below 1.3 times the computed stall speed (V_S). In this mode of operation, the flight control computer will modify thrust in order to maintain an angle of attack to control the aircraft to an airspeed of greater than 1.3 V_S, with gust compensation included to increase the desired airspeed to compensate for wind conditions. The stall margin mode is also engaged when the flaps are deployed to greater than 30°.

The thrust management mode is engaged to control the throttles in response to commands from an external computer (not part of the FCS-240). The mode is selected by pressing the dedicated pushbutton of the glareshield panel.

The flare mode is a programmed reduction of airspeed upon acquisition of the flare mode. The flight control computer will reduce thrust in order to reduce airspeed at a rate of 0.9 knots/sec.

Subsequent to the flare mode upon descending through 5 ft or main strut compression, the flight control computer will retard the throttle to the ground idle position at a rate of 8°/sec of throttle lever motion, after which the autothrottle will be disengaged automatically.

2.1.6.3 The Sixty Program

The sixty program is executed 60 times a second or every 16.7 msec. The algorithms included are limited to those related to the longitudinal inner-loop control which require a very high computational rate such as pitch rate feedback, pitch rate command limiting, and normal acceleration damping during autoland.

2.1.6.4 Error Service Routine

The following events trigger an interrupt which causes control transfer to the error service routine:

Bus time out (attempt to transfer to/from nonexistent address)
Illegal operational code
Stack overflow
Arithmetic overflow

The error service routine decodes a status word to determine the cause of the interrupt and then stops the program execution by transferring control into a tight endless loop. In service this condition produces a servo disconnect and can only be cleared by cycling power.

2.1.6.5 Background

The background program consists of the following three procedures which are always scheduled and executed in the following order: self test, CPU diagnostic, and fault isolation.

The self-test program includes the following modules: CK sum test, NE monitor, and GSP test.

The CK sum test performs one test for each PROM card in the system by comparing the bit-check sum of each card with the value prestored in the two top addresses of the same card. This module is executed at least every 20 sec.

NE monitor is primarily an execution order detector which checks that the software executes properly within the allocated main frame time of 50 msec. This is accomplished by (1) toggling a discrete every 50 msec (the hardware will disconnect if no toggling occurs within a 20% time tolerance) and (2) incrementing and comparing

two separate counters at the top and bottom of the foreground to assure that the foreground execution is complete every time that foreground is initiated. NE monitor also tests the validity flags from all the sensors.

GSP test module monitors the glareshield panel by wrapping around unique bit patterns to the GSP via the Arinc 429 bus.

The CPU diagnostic, the only procedure within the FCC written in assembly language, tests the proper execution of all the instructions and by doing so updates the content of a counter which represents the total number of instructions tested. A failure condition occurs if:

1. Any instruction fails to execute properly.

2. The content of the counter, at the end of the diagnostic program, is not set to the correct value.

3. The CPU diagnostic program fails to terminate within 40 sec.

The fault isolation procedure detects malfunctions of systems components other than the FCC itself, such as servos, sensors, etc., and isolates the malfunctions to line replaceable unit levels. This information is then transmitted to the fault isolation and data display system (FIDDS) (not included in the DFCSVL).

2.2 CAPS TEST ADAPTER

Each of the four CAPS transfer buses (one for each processor of the FCS-240) is connected to a CAPS test adapter (CTA). Each CTA is dedicated to one processor and allows the operator access to the associated CAPS transfer bus directly from its front panel controls (fig. 11) or from the Hewlett Packard terminal included in the pallet. The capabilities that the CTAs provide, similar to those of the operators console of many commercial minicomputers, are listed below.

1. Display of transfer bus address and data in hexidecimal or binary.

2. Examine and modify any bus-addressable location.

3. Single bus-step or single instruction-step.

4. Halt the processor when a preselected address is accessed, when that address has preselected data, or if a read/write condition occurred.

5. Monitor the contents of a selected address during dynamic operation.

6. Record the 16 most recent transfer bus read or write cycles. Data stored are address, data, and status.

7. Perform continuous conversion of four selected addresses through a 12-bit D/A converter with four separate front panel outputs available for strip chart recording. A HI/LO switch is provided so that either the 12 high-order or the 11 low-order plus sign bit of the 16-bit CAPS bus may be monitored.

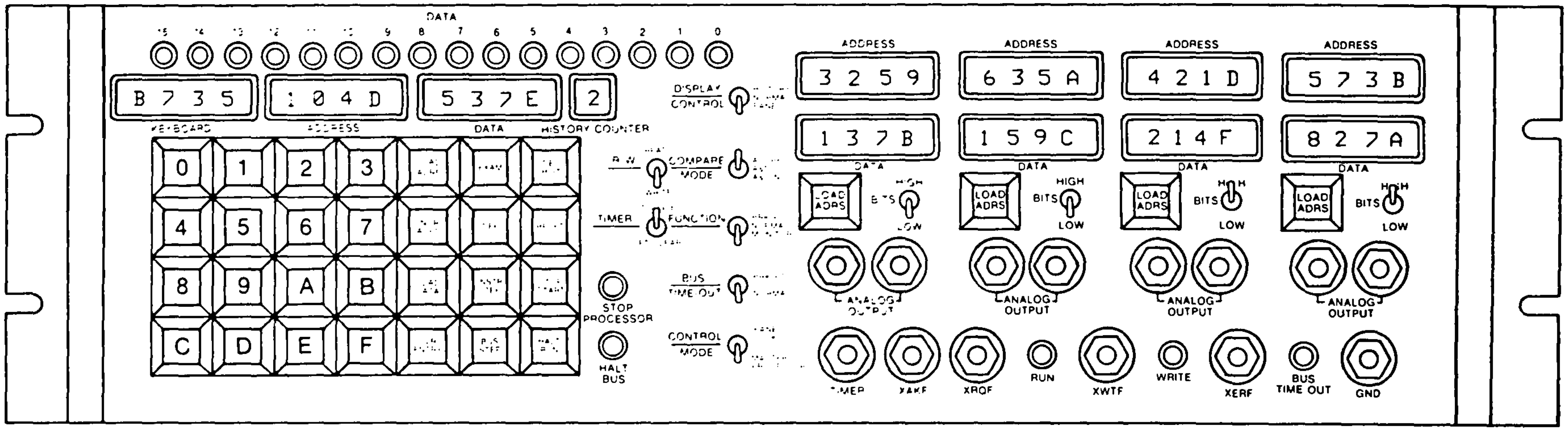


Figure 11.- CAPS test adapter front panel.

Each CTA has an interbus channel (see fig. 12) which creates a link under control of the PDP-11/04, included in the pallet, between the UNIBUS and the CAPS transfer bus. This link effectively makes the transfer bus an extension of the PDP-11/04 UNIBUS; each CTA appears to the PDP-11/04 as eight contiguous UNIBUS 16-bit word addresses.[1] These eight UNIBUS addresses allow PDP-11/04 access to the following CTA components:

Component	Type of access
CONTROL/STATUS REG	R/W
ADDRESS DISPLAY REG	R
DATA DISPLAY REG	R/W
ADDRESS HISTORY PORT	R
DATA HISTORY PORT	R
KEYBOARD DISPLAY REG	R/W
TARGET ADDRESS REG	R/W
TRANSFER BUS WINDOW	R/W

The transfer bus which interfaces the CAPS with the CTA includes bus control, address, and data lines. The CTA acts as bus master only during the following:

HALT/RUN requests
Breakpoint states
Time-out bus-error interrupts
CTA initialed memory read/write operations

In all other modes the CTA acts as a passive monitor of transfer bus address and data lines without adding to CPU or transfer bus overhead (loading).

2.3 MODULAR DIGITAL INTERFACE CONTROL UNIT

The modular digital interface control unit (MDICU) is a programmable, CAPS-6 based data distributor whose primary function is the control of the flow and format of the simulated aircraft parameters generated by a PDP-11/60, external to the pallet, and of the control commands generated from the flight computers. This function enables the closed-loop operation between the flight computers in the pallet and the aircraft simulation in the PDP-11/60.

The data conversion applied to each parameter is shown in table 7. The rate of data conversion is independent of and asynchronous with respect to all other system elements. Data transport delays are within acceptable limits since the MDICU cycle rate is typically 5 times the flight computer cycle rate, depending on tradeoffs between software and hardware processing within the MDICU.

A total of 99 I/O parameters can be processed by the MDICU. Table 8 shows the parameters processed by the MDICU in the current configuration. The following limitations apply:

---

[1]The HP terminal provides the user interface to the PDP-11/04.

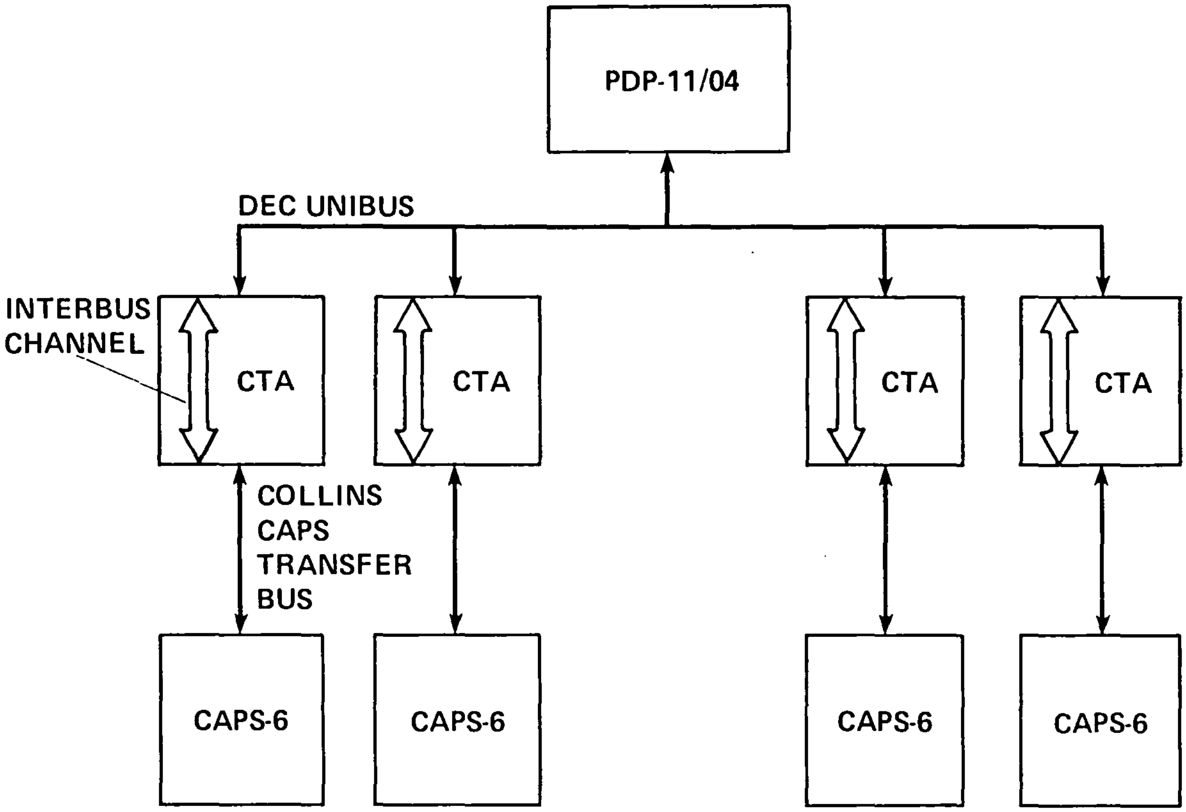


Figure 12.- CAPS test adapter interconnection.

TABLE 7.- CONVERSION CHANNELS

Converter type	Number of channels currently assigned	Number of spare channels	Data Source
A/D	6	10	FCC
D/A	22	1	11/60
Dig/synchro	3	13	11/60
Dig/2 WAC	3	9	11/60
Dig/dis	0	32	--
Arinc 429	3	0	11/60

TABLE 8.- CONVERTED VARIABLES

Data to MDICU from 11/60		Data from MDICU to 11/60
TAS	PITCH RATE	AILERON POSITION
MACH	MAG HDG	RUDDER POSITION
IAS	LAT ACCEL	STABILIZER POSITION
BARO ALT RATE	LONG ACCEL	DIRECT LIFT CONTROL
BARO ALT CORR	RAD ALT	% THRUST
BARO ALT	G/S DEV	One word reserved for SAC discrete
YAW RATE	NORM ACCEL	
VOR BRG	STATIC AIR TEMP	
AOA	LOC DEV	
FLAP	ROLL ATT	
PITCH ATT	RADIO ALT	
Ten parameters repeated for routing to SAC		

(a) Number of parameters from 11/60 to MDICU ≤ 64

(b) Number of parameters from MDICU to 11/60 ≤ 64

(c) (a) + (b) ≤ 99

The HP 2645 terminal provides direct operator control over the operation of the MDICU to monitor program execution, insert faults, and allow normal processing of each parameter. The following specific capabilities are possible:

1. Load/dump programs between MDICU memory and HP tape cartridges.

2. MDICU program modification (patching) from HP keyboard.

3. Total control of data conversion type and scaling applied to each parameter.

4. Manual read/write operations on MDICU memory during static or dynamic operation.

5. Initialization of all parameters to a selected value as an initial condition.

6. Application of bias, step, or ramp operations to any single parameter.

These capabilities provide a closed-loop real-time simulation environment with a full complement of fault-free or selectively faulted processes.

Figure 13 shows the major MDICU elements. Aircraft state parameters generated by the PDP-11/60 are written into a 64-word scratch pad memory within a serial I/O card. Each parameter is then read by either the MDICU software or by a hardware DMA card, depending on prior specification in the program. Irrespective of the preselected option, the data are read from assigned unique scratch pad address, scaled, limited, formatted, and routed to an output data converter directly wired to a flight computer input port. Conversely, command outputs from the flight computers are directly wired to A/D input converters within the MDICU. The command values are read from the unique address occupied by the A/Ds, scaled, and routed to the assigned scratch pad address on the serial I/O card for transmission to the PDP-11/60.

The serial I/O card in the MDICU communicates with a similar card in the 11/60 over a serial, Manchester coded, asynchronous link under control of the transmitting portion of the respective card. No host processor (MDICU CAPS-6 or 11/60) control is utilized once the 16-bit data word is installed into the scratch pad RAM (SPR) memory. The transmit sequence is as follows: the card sequentially reads from its SPR, attaches a 6-bit predetermined destination address to each data parameter, converts from parallel to serial, appends a parity bit, Manchester encodes the serial word, and then under its internal hardware control sends the resultant 23-bit word over the serial link to the 11/60. The receive sequence is as follows: the incoming 23-bit word from the corresponding 11/60 serial I/O card is Manchester decoded, checked for parity, converted from serial to parallel, and the 16-bit data word is stored in SPR according to the 6-bit address which was attached by the 11/60 serial I/O card.

Once the received word is stored in SPR it is available to either the MDICU CAPS-6 or DMA hardware for processing. Contention logic prevents any concurrent READ-WRITE conflicts from occurring at the same location for both transmit and receive portions of the card.

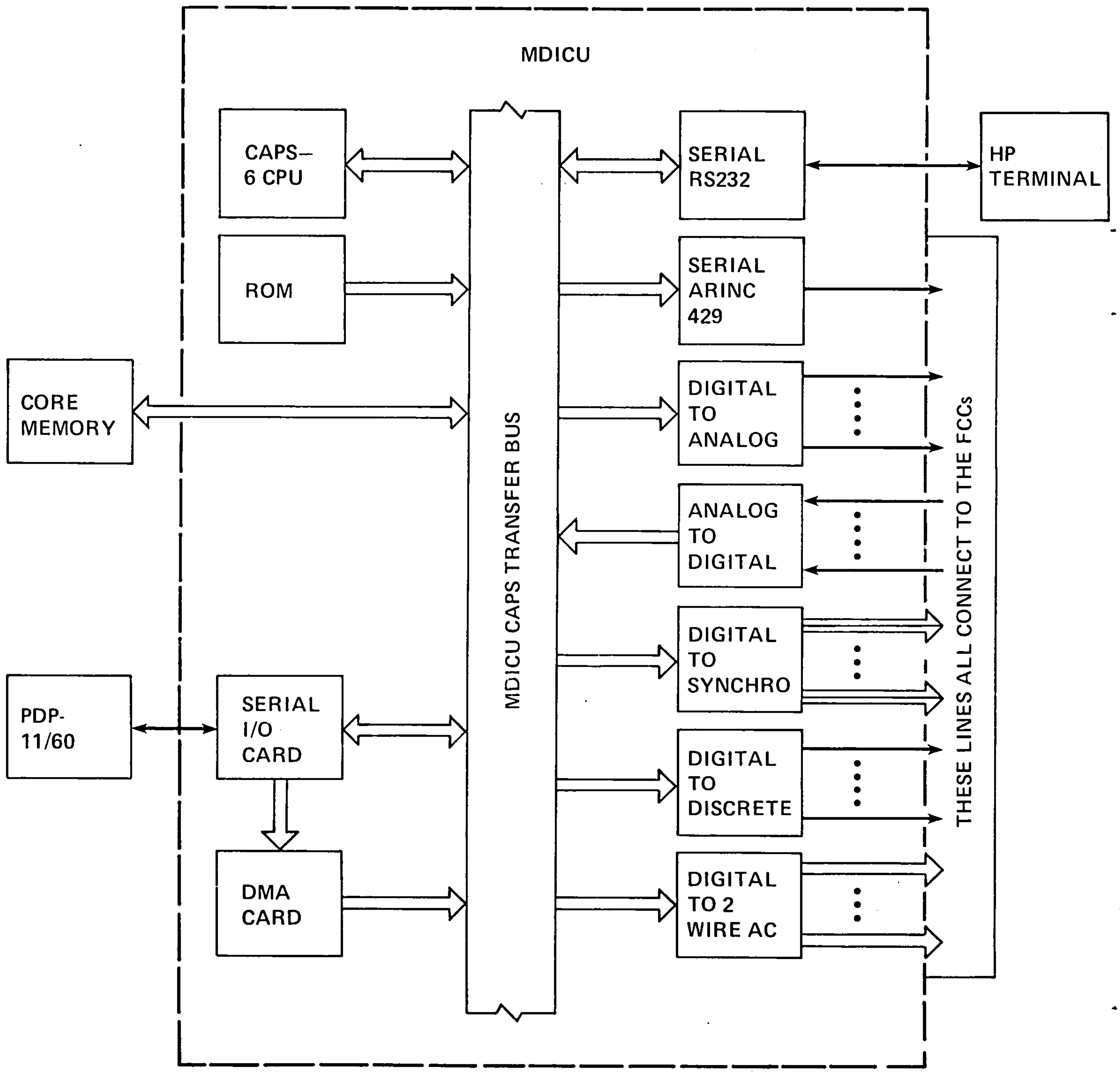


Figure 13.- MDICU functional block diagram.

The MDICU has a memory-mapped I/O structure wherein each SPR location and data converter channel occupies a unique CAPS-6 transfer bus location. Therefore, the I/O address, whether contained in software or DMA hardware, specifies the SPR location and data converter combination to READ from and WRITE to for each sensor (state) parameter or flight computer command processed by the MDICU. The specific addresses of all data processed by the MDICU are contained in the "PDP-11/60-MDICU Interface" document.

The control of each parameter/command received or transmitted is defined by the contents of its assigned "BEAD" in the MDICU software. A BEAD is a table read by the main program which defines hardware or software control, scaling, initial condition values, input address, and output address(es) for any single variable. This BEAD is supported by the AED language as a structural element. This allows the capability to easily change and customize the sensor data to fit various user needs. The main program is organized as a continuous loop which sequentially interprets the BEADS as it controls parameter flow. The present MDICU software occupies 22K words of the 24K memory available.

A significant speed advantage results from specifying parameters to be under DMA hardware control. The DMA card operates in parallel to the main program and directly reads and writes from input addresses to output addresses depending on MDICU CAPS-6 transfer bus availability. In our present configuration a 50-Hz MDICU cycle rate results under all software control, while a 150-Hz rate is possible with total DMA control. Exclusive DMA control of all parameters is generally not used since flexibility is sacrificed in the ability to insert failures to the parameter processing via the HP terminal.

2.4 SERVO SIMULATOR

The servo simulator (S/S) receives the pitch, roll, and yaw commands from the FCCs and utilizes analog circuitry to simulate the corresponding modulation pistons and power servos response. The S/S also receives the FCC autothrottle commands and provides tachometer feedback to the FCCs. The surface positions and a derived thrust signal are routed from the S/S to the MDICU for transmission as inputs to the aircraft model on the PDP-11/60 and to the FCC to close inner-control loops.

The S/S has the capability to insert the following faults by using controls on the front panel (see fig. 14):

1. Pitch, roll, and yaw coil currents failure as HARDOVER, OPEN, or SLOWOVER of either positive or negative values for ECC1.

2. Pitch trim failure to HARDOVER.

3. Pitch, roll, or yaw modulation piston feedback fault signal values of selectable amplitude.

The S/S provides appropriate monitoring features via a panel-mounted DVM and output jacks with which the operator may observe fault specifics.

The front panel controls include a potentiometer, labeled J-curve, used to select a linear operating region within the nonlinear pitch power servo transfer function.

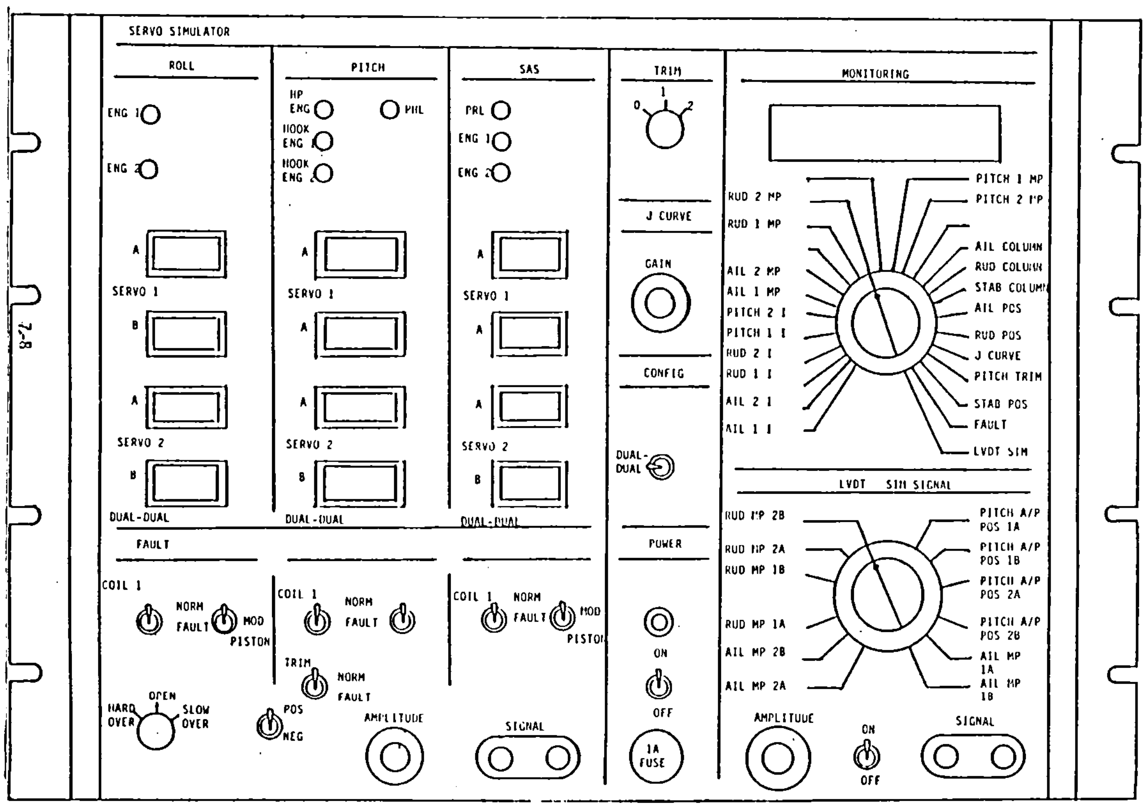


Figure 14.- Servo simulator front panel.

Figure 15 shows a typical simulated servo. The FCC commands are output as coil currents to the modulation piston, simulated as a first-order lag, where they are terminated in an appropriate resistive load. Front-panel meters continuously monitor the currents. The modulation piston portion is fed back to the FCCs and to the power servo, also simulated as a first-order lag, to generate the control surface position. The MDICU then converts the surface position signal into a synchro signal, which is routed to the FCC and to the PDP-11/60. The transfer functions are established by dedicated hardware circuits within the S/S.

The S/S also contains circuitry which emulates the engage logic for pitch, roll, and yaw with front-panel LED indicators for the corresponding status.

There is a pitch trim circuit in the S/S which receives and holds the trim command from the FCC. This trim is input to the power servo section of the S/S so that the modulation piston only supplies dynamics. The operator can optionally select via the front-panel switch several constant pitch trim rates. The roll and yaw trims are input from the FCC as part of the axis modulation piston coil currents.

2.5 GLARESHIELD PANEL

The primary pilot-control interface is through the glareshield panel (GSP). Figure 16 shows and labels the controls and displays on the face of the GSP. Table 9 is a tabulation of the function of each item shown in figure 16. The primary function of the GSP within the Verification Laboratory is the establishment of the various flight modes available from the RDFCS. The GSP electrical interface with the FCC is via standard ARINC 429 serial data link.

2.6 BREAKOUT PANEL

There are two breakout panels contained in the pallet. These panels consist of terminal strips where all signals entering or leaving the rear connectors of the FCC are routed. The panels contain bottle plugs which when inserted into the panel complete the signal path and when removed cause an open in the signal path. Along with these bottle plug points are other parallel contacts. The primary functions of the breakout panels are for signal monitoring and fault introduction.

2.7 DISCRETE SWITCH PANEL

Figure 17 shows the discrete switch panel. It is used to input the discrete valid conditions necessary for the various modes of the FCC. As can be seen from the figure the valids represent those from subsystem units that normally interface to the FCC when installed in an aircraft. The availability of these valids/discretes at individual toggle switches allows the user the capability of enabling or disabling each signal independently. This is a convenient method of manually introducing the types of sensor/subsystem faults that are "flagged" by their corresponding valids.

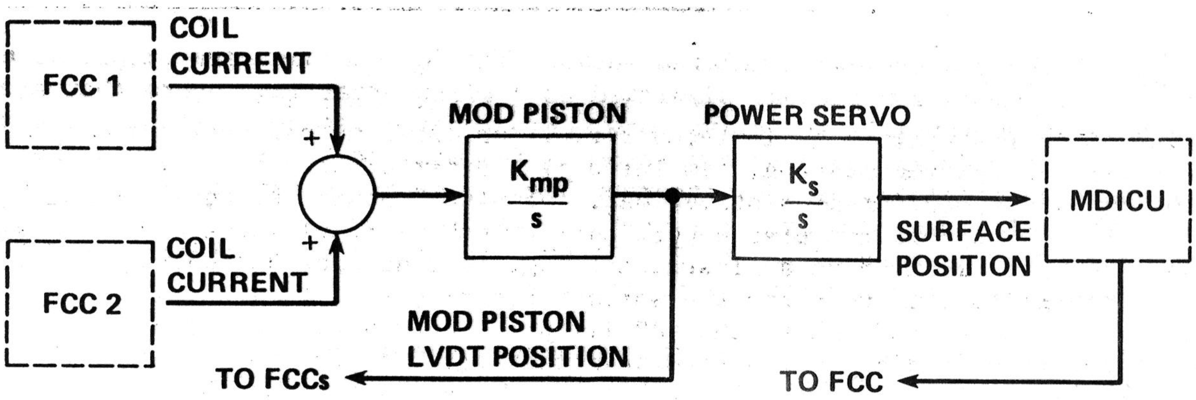


Figure 15.- Servo model.

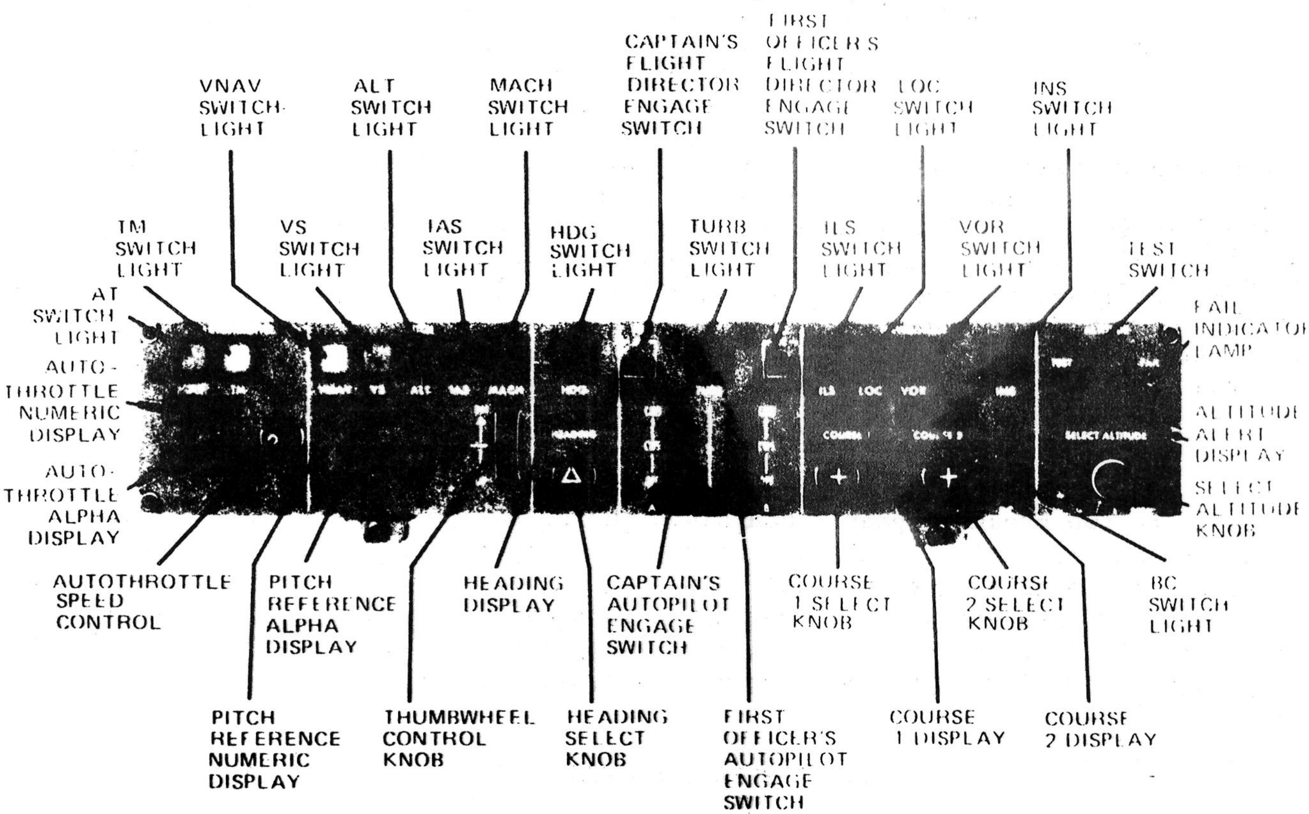


Figure 16.- Glareshield panel.

TABLE 9.- GLARESHIELD CONTROLS AND FUNCTIONS

Control/indicator	Function
AUTOTHROTTLE AT switchlight (S1)	Provides selection and annunciation of AUTO-THROTTLE IAS (indicated airspeed) hold mode.
TM switchlight (S2)	Provides selection and annunciation of TM (thrust management) mode.
AUTOTHROTTLE numeric display (A1)	Provides numeric indication of selected airspeed in knots when AUTOTHROTTLE system is in IAS hold mode.
AUTOTHROTTLE alpha display (A5)	Displays 3-character mnemonic representing the AUTOTHROTTLE mode engaged.
AUTOTHROTTLE speed control (S21)	Allows adjustment of engaged AUTOTHROTTLE function, or airspeed if no modes are engaged.
VNAV switchlight (S4)	Provides selection and annunciation of VNAV (vertical navigation) mode.
VS switchlight (S5)	Provides selection and annunciation of VS (vertical speed) hold mode.
ALT switchlight (S6)	Provides selection and annunciation of ALT (altitude) hold mode.
PITCH IAS switchlight (S7)	Provides selection and annunciation of PITCH IAS (indicated airspeed) hold mode.
MACH switchlight (S8)	Provides selection and annunciation of MACH (Mach number) hold mode.
PITCH reference numeric display (A7)	Provides numeric indication of selected vertical speed, or actual altitude, airspeed, or Mach number, contingent on which mode is engaged.
PITCH reference alpha display (A6)	Displays 3-character mnemonic representing the PITCH mode engaged.
Thumb-wheel control knob (S22)	Allows adjustment of vertical speed when VS hold mode is engaged.
HDG switchlight (S9)	Provides selection and annunciation of HDG (heading) select mode.
HEADING display (A2)	Provides numeric indication of aircraft heading in degrees.
HEADING select knob (S23)	Selects heading of aircraft when HDG select mode is engaged.
Captain's FD engage switch (S19)	Engages (ON) or disengages (OFF) captain's flight director system.
TURB switchlight (S10)	Provides selection and annunciation of TURB (turbulence) mode.
First officer's FD engage switch (S20)	Engages (ON) or disengages (OFF) first officer's flight director system.

TABLE 9.- Concluded.

Control/indicator	Function
Captain's autopilot engage switch (S29)	Engages captain's autopilot into the basic CWS (control wheel steering) configuration or the CMD (command) configuration. Disengages autopilot in the OFF position.
First officer's autopilot engage switch (S30)	Engages first officer's autopilot into the basic CWS (control wheel steering) configuration or the CMD (command) configuration. Disengages autopilot in the OFF position.
ILS switchlight (S11)	Provides selection and annunciation of ILS (instrument landing system) mode.
LOC switchlight (S12)	Provides selection and annunciation of LOC (localizer) mode.
VOR SWITCHLIGHT (S13)	Provides selection and annunciation of VOR (vhf omnidirectional and radio range) mode.
INS switchlight (S15)	Provides selection and annunciation of INS (inertial navigation system) mode.
COURSE 1 display (A3)	Provides numeric indication of selected no. 1 (captain's) course in degrees.
COURSE 1 select knob (S24)	Selects no. 1 (captain's) course.
COURSE 2 display (A4)	Provides numeric indication of selected no. 2 (first officer's) course in degrees.
COURSE 2 select knob (S25)	Selects no. 2 (first officer's) course.
BC switchlight (S17)	Provides selection and annunciation of BC (back-course) mode.
TEST switch (S28)	Allows self-testing of altitude alert system when aircraft is on the ground.
FAIL indicator lamp (DS1)	Indicates failure of altitude alert system during self-test.
SELECT ALTITUDE display (A8)	Provides numeric indication of selected altitude for alert system and automatic altitude capture arm.
SELECT ALTITUDE knob (S26)	Selects altitude for alert system and automatic altitude capture arm.
Normal/standby select knob (S27)	Selects no. 1 (normal) or no. 2 (standby) air data system as the reference.

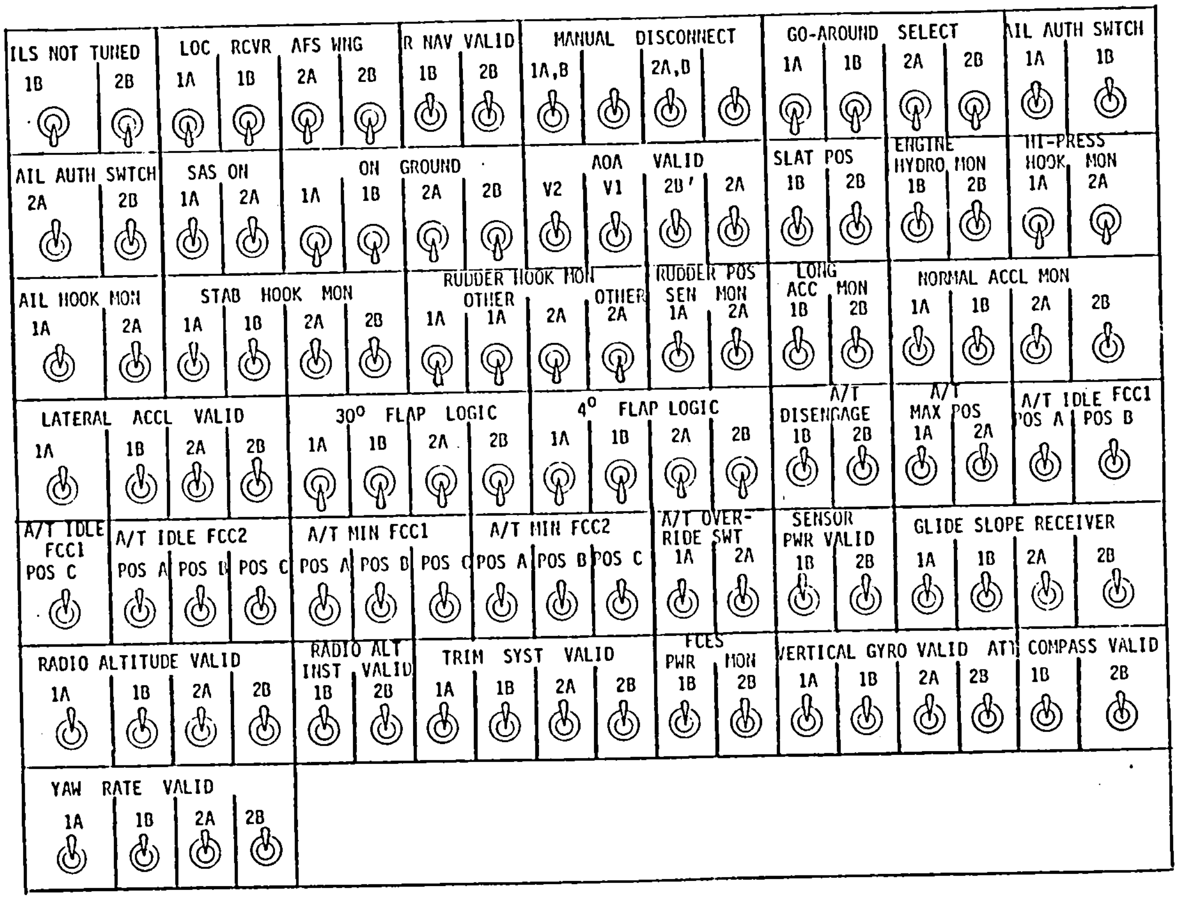


Figure 17.- Discrete switches.

2.8 BUFFER PANEL

The buffer panel (BP) provides two basic functions: one to break out or make accessible some internal test points of the FCC-201 computers, the other to provide a signal buffering capability to isolate analog and discrete pallet signals available at the various breakout panels from the laboratory recording equipment.

To use the test point breakout capability, the BP must be connected to the FCC-201 computers via 40 conductor ribbon cables and test-access extender cards that plug into the FCC-201 test-access slots. A total of 160 test points are available from each FCC-201 (80 per channel).

To use the buffer section the signal of interest must be connected via jumper wire from its respective BP location to an input jack on the BP front panel. There are 32 analog buffers and 24 logic buffers. The outputs of all the analog and logic buffers connect to a 61-pin circular connector (P9) located on the rear of the BP. The interface with strip-chart recorders or other instrumentation is via this P9 output connector.

The analog buffers are unity gain, differential input op-amp circuits. Each op-amp output goes to a pin on P9. Each op-amp input goes to a jack on the front panel and can be used in either a differential or single-ended configuration. When used single ended an inverting or noninverting mode is selectable.

The logic buffers can accept either a TTL or +28 V input level and provide +28 V or open output levels to the recorder. The buffer inputs must be patched via wires into jacks on the BP front panel. Only the high side of each buffer input is patchable. The low side is referenced to chassis ground.

2.9 OTHER FLIGHT INSTRUMENTS

The pallet also contains flight instruments that display the information generated by the FCC. Figures 18, 19, 20, and 21 show the warning annunciator, mode annunciator, ADI, and HSI, respectively. These figures are self-explanatory as to function and capability. However, for a detailed understanding of a particular selected mode displayed upon these instruments the reader is referred to the FCC section of this publication.

2.10 PDP-11/04

The PDP-11/04 is used as an interface between the PDP-11/60 or the HP-2645A and the FCC. The PDP-11/04 combined with the HP-2645A terminal can duplicate all the functions of the CTA. Additionally, the HP-2645A terminal features memory storage which can be used to display on a screen or dump on a line printer or magnetic tape cassette blocks of FCC memory. These blocks of memory can also be transferred and stored in the PDP-11/60 through the PDP-11/60 — PDP-11/40 — FCC link.

The functions supported by the PDP-11/04 include:

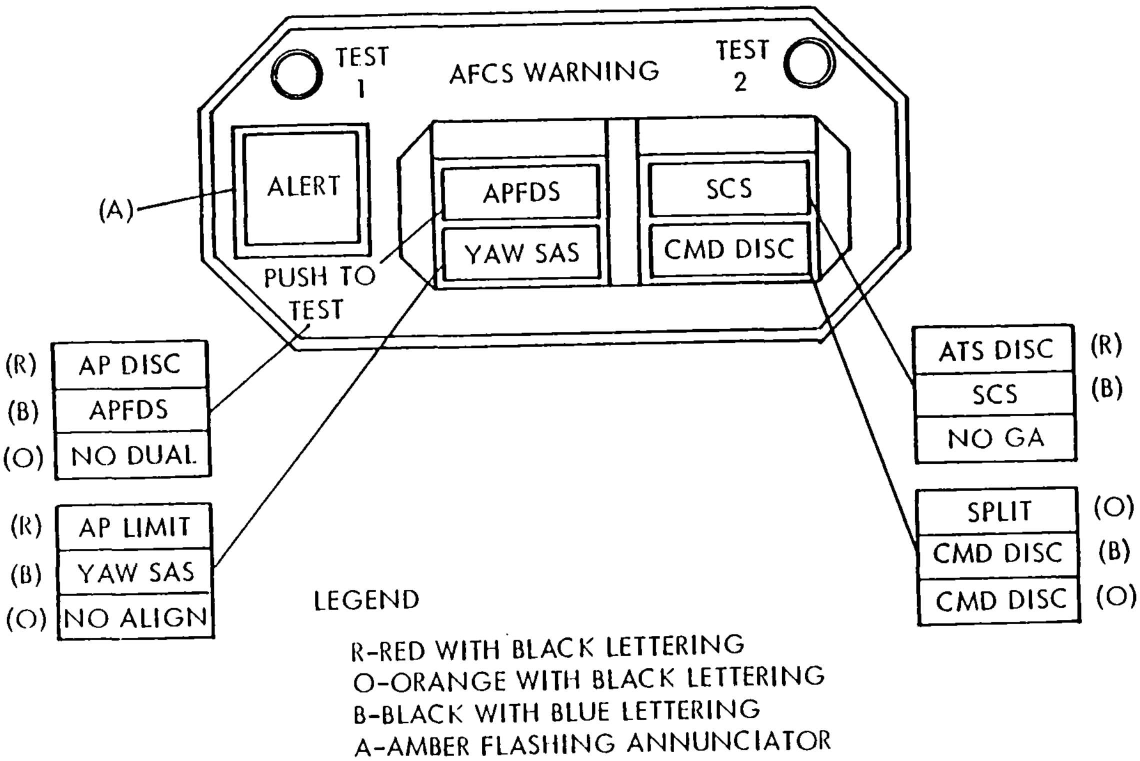


Figure 18.- Warning annunciator.

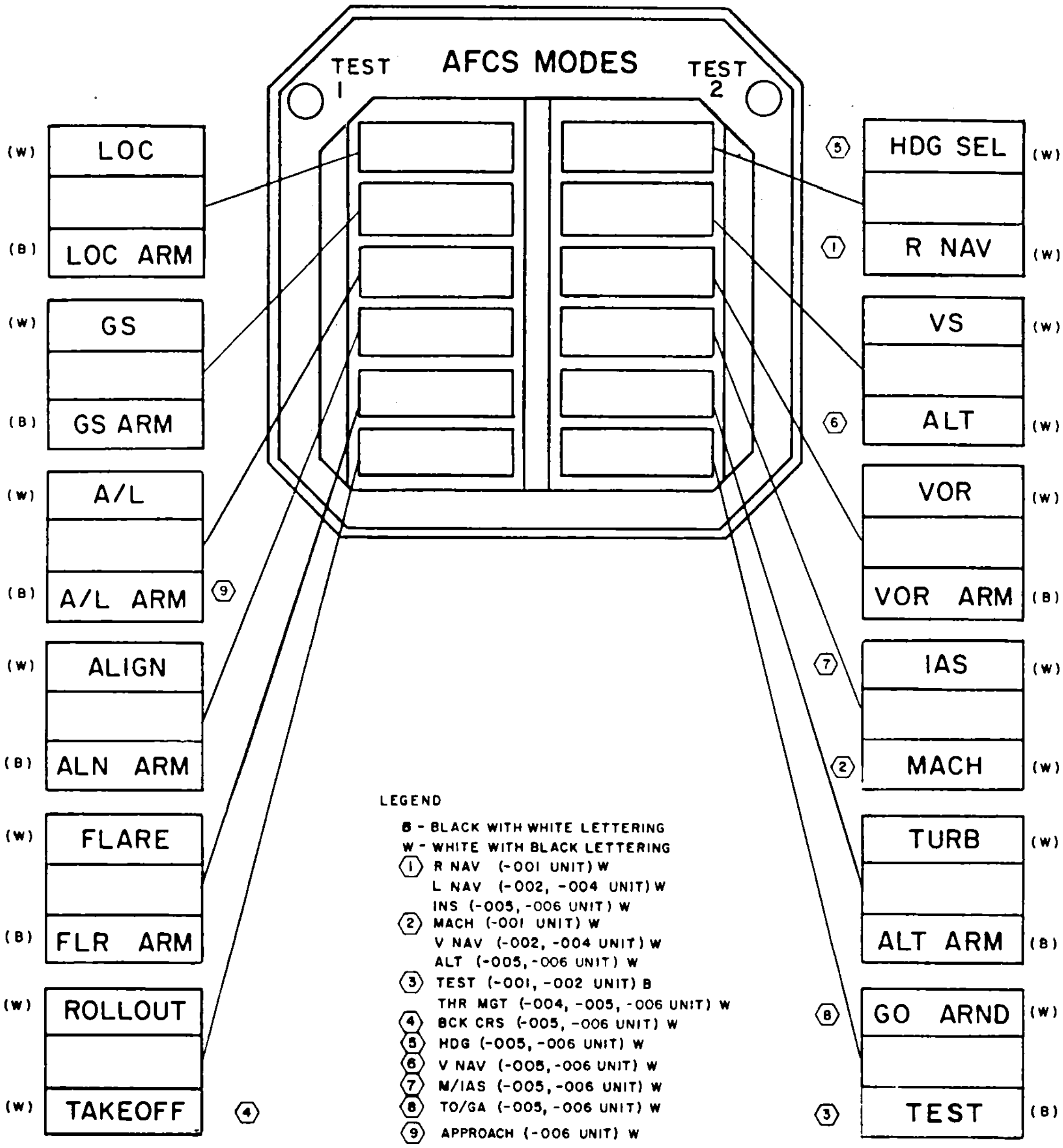


Figure 19.- Mode annunciator.

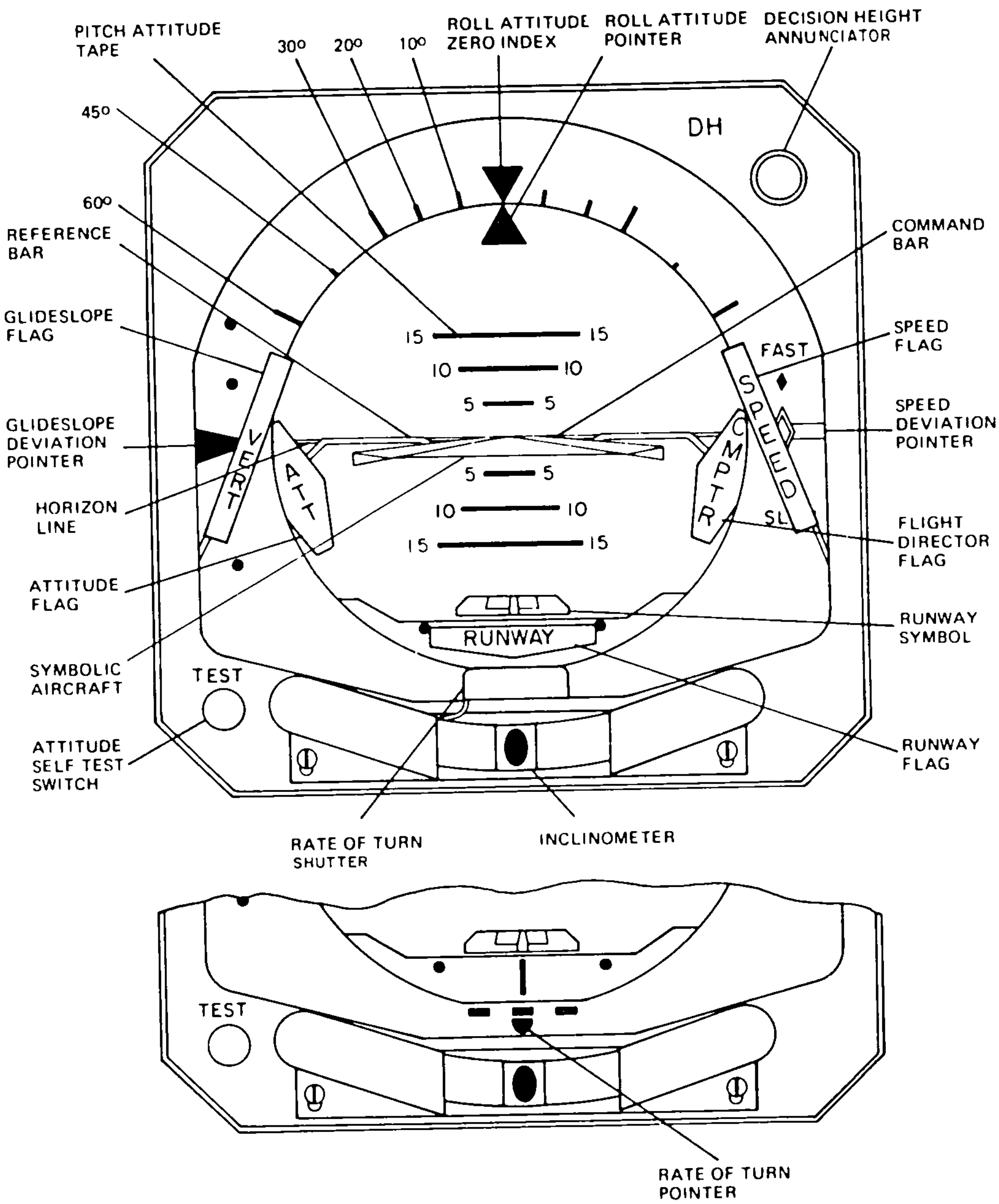
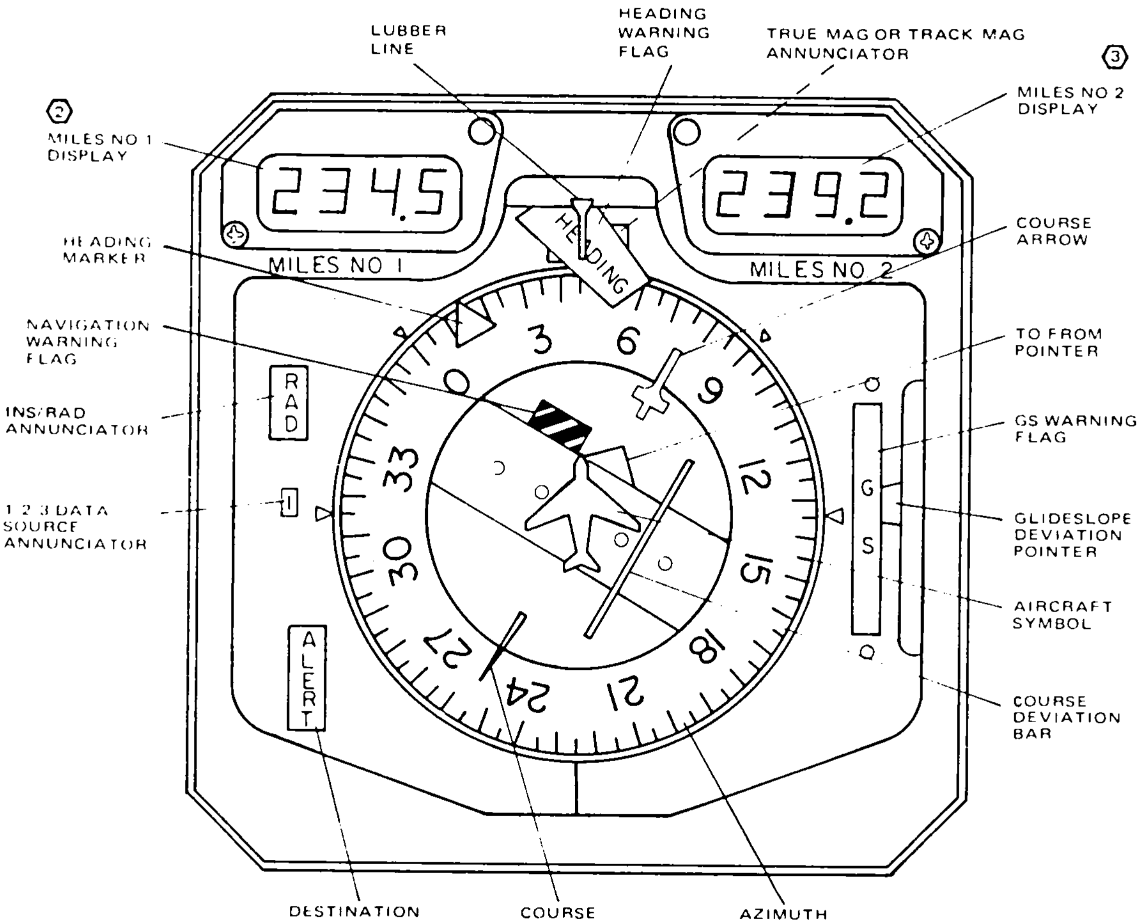


Figure 20.- Attitude director indicator.



NOTES

① THE INTERMITTENT FAULT INDICATOR IS LOCATED INSIDE THE HSI.

② DISPLAY IS LABLED MILES ON (CPN 772 5025-011) AND GROUND SPEED ON (CPN 772-5025-012).

③ DISPLAY IS LABLED MILES ON (CPN 772 5025-012) AND GROUND SPEED ON (CPN 772-5025-011).

Figure 21.- Horizontal situation indicator.

1. All functions which can be performed by the CTA. These functions have been discussed in the section covering the pallet.

2. Uploading and downloading blocks of FCC memory into internal devices and PDP-11/60. The operation can be performed with the FCC in a halt mode or while the flight software is being executed in real time.

Two data communication interfaces are utilized by the PDP-11/04:

1. A DR11-B general purpose, direct memory access (DMA) interface between the PDP-11 UNIBUS and the CTA.

2. A DA11-B interprocessor link which establishes a DMA, parallel data transfer channel between the PDP-11/60 and PDP-11/04. Sixteen bits of parallel data can be sent or received in word or block mode transfers. Maximum block length is 32,768 words but the software is currently designed for block transfers of 512 words.

The interconnections between the PDP-11/04 and the other elements of the pallet are shown in figure 22.

3. STAND ALONE CAB

The stand alone cab is a modified flight trainer, the ATC 510. For this application its analog simulation circuitry was replaced with special interface circuits designed to send and receive analog data to and from the pallet. It provides the capability to send manual pitch, roll, and yaw inputs to the servo simulator in the pallet and receive aircraft state data from the MDICU (generated in the PDP-11/60) for display on the pilot instrument (see fig. 23). While the stand alone cab is not intended to provide a high-fidelity pilot environment it does provide the user with the capability to manually control the aircraft simulation and to observe the corresponding model outputs. When the standard Collins instruments previously described are combined with the stand alone cab the environment is enhanced and will support a limited variety of pilot-in-the-loop experiments.

4. UNIVAC 1100

The UNIVAC 1100 is a main-frame computer, located off-site and connected to the PDP-11/60 through a dataphone link. It hosts the AED processors and V&V software tools. The UNIVAC runs under the control of the EXEC-8 operating system and is fully compatible with the UNIVAC 1100 system which hosts the validated AED processors used for developing programs of critical DFCS.

4.1 REMOTE LINK

The UNIVAC 1100 is connected to the PDP-11/60 through a dataphone link and permits the PDP-11/60 to function as a remote job entry station. The dataphone is a 4800 model 208B data set and is designed for the transmission and reception of data at 4800 bits/sec.

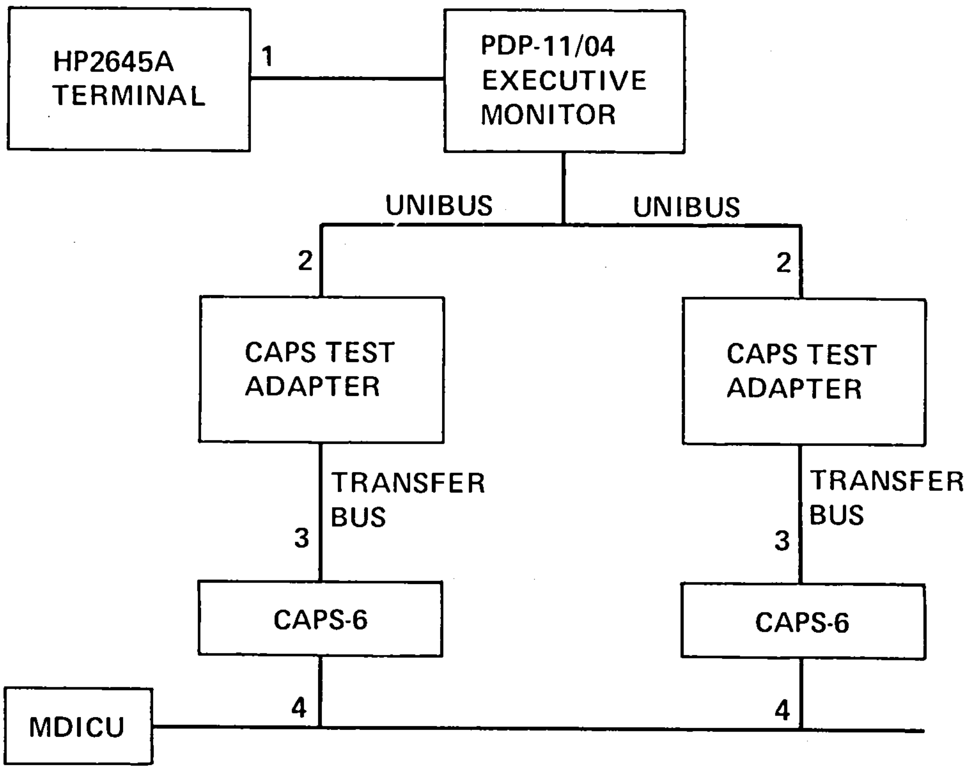


Figure 22.- PDP-11/04 Interface.

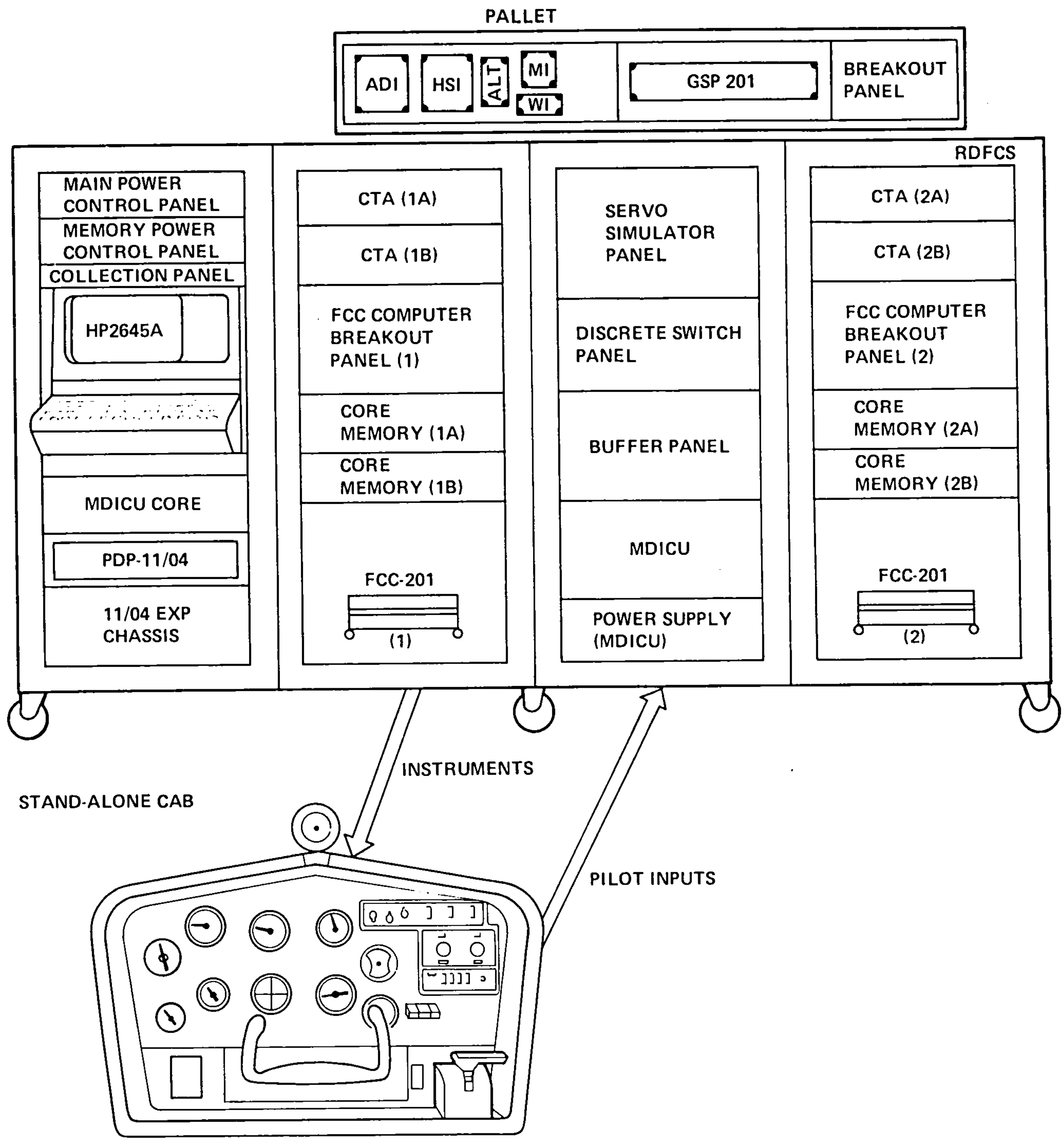


Figure 23.- Stand alone cab-pallet interface.

A communications link can be transparent or nontransparent in its transmission. A fully transparent communications link transmits data in any form and the data are received exactly as transmitted. A nontransparent communications link performs some data manipulation prior to the transmission. The dataphone link for this system is a nontransparent "HASP" (for this reason, after the transmission to the PDP-11/60 object files and load modules must be converted again to the original format prior to any further utilization).

A secondary communications channel to the UNIVAC 1100 is provided by a direct link with the HP-2645A terminal.

4.2 AED SUPPORT SOFTWARE

The AED support software include the AED cross-compiler, the AED assembler, the AED link editor, and a tape transmission program. These programs are used to produce critical flight software.

4.3 V&V SOFTWARE TOOLS

An integrated set of software V&V tools is hosted in the UNIVAC 1100. These tools support software testing by (1) enhancing the effectiveness of the conventional closed-loop real-time simulation environment and (2) providing additional testing capabilities. The tools which require minimum or no manual intervention can be broadly divided into three groups — static tools, dynamic tools, and documentation tools.

4.3.1 Static Tools

Static tools do not require the flight software to be executed. They check for semantic errors over all paths of the programs.

A brief synopsis of some of these static tools is listed below.

1. Set/use— This tool checks for local and global variables which are never set, or not set in some path, or set but not used. These variables will be automatically flagged during static consistency analysis.

2. Infinite loop— This tool checks for loops with no exit, DO loop index used after loop, uninitialized loop variables, and nonmonotonic loops.

3. Unit assertion— This tool checks that the physical units of both sides of an assignment statement are consistent with the declared physical units of each variable.

4. Symbolic executor— This tool provides the capability to symbolically execute AED expressions over specified program paths. The output of the symbolic executor can be used to validate the correct implementation of algorithms.

4.3.2 Dynamic Tools

Dynamic tools add instructions to the source code to provide enhanced code-testing capabilities. This operation is often called code instrumentation. After the instrumented code is compiled and executed, the dynamic tools provide a series of reports with information of program execution and behavior. Two major capabilities of these tools are described below.

1. Logical and timing assertion— Timing of module entries and exits are automatically computed and recorded. Execution time exception reports are also generated for all user-supplied timing assertions.

The logical assertions describe conditions which are true at the point of assertion. Error reports are generated whenever the assertions are violated.

2. Path instrumentation— The instrumentor for dynamic analysis provides program path instrumentation. After the instrumented program has been executed, reports which show path coverage are automatically generated.

4.3.3 Documentation Tools

The automated documentation generator provides several documentation reports at the module level or at the program level. The following is a description of some of these reports.

1. Indented listing— A source listing is automatically indented according to the control structure of the program.

2. Cross reference— Reports for symbols, labels, global and external references are provided. For example, the symbol cross-reference report will include all the symbols, the scope of the symbol, the module(s) of reference, statement number of reference, and a flag denoting definition, usage, or value assignment of the name.

3. Reaching set/calling tree— The reaching set is a source text which shows, within a module, all the branches which must be executed prior to reaching a specified statement.

The calling tree is a hierarchical representation of module interaction, including notation for modules which are external, non-nested, nested, re-entrant, and recursive.

5. PDP-11/60 — THE ENVIRONMENT COMPUTER

The PDP-11/60 is the central element of the DFCSVL (fig. 24). It is fundamental to all the operating scenarios possible within the laboratory since it integrates the other laboratory elements into an efficient user-oriented environment for the verification of digital flight control systems. From a single terminal connected to the PDP-11/60 (the INtext terminal) the user can interface to and control all the resources of this environment.

The PDP-11/60 supports two distinct environments:

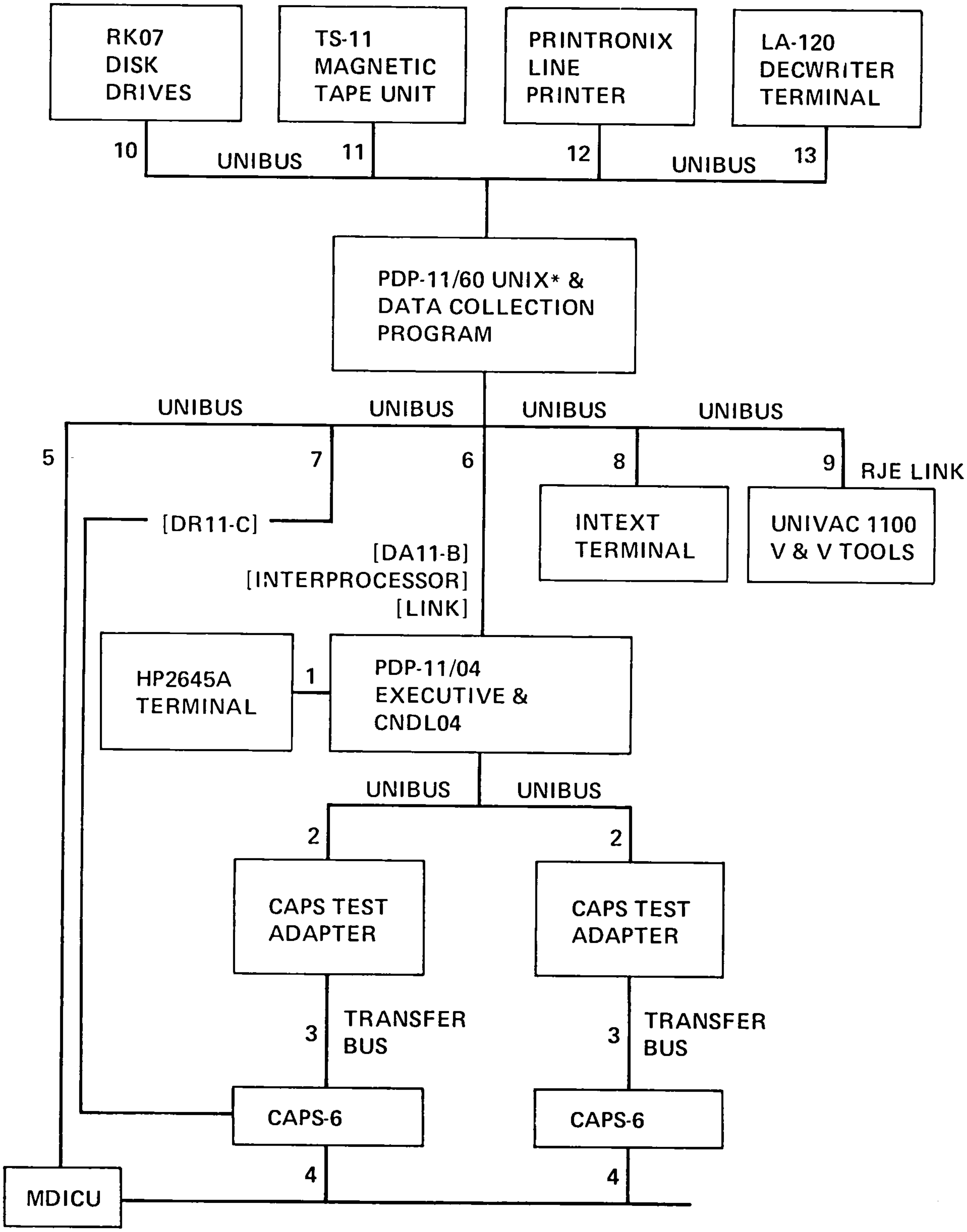


Figure 24.- DFCS facility.

1. A code-development environment to develop new code, edit, compile, and assemble modules and generate load modules. This environment also supports verification with the automated tools hosted in the UNIVAC processor.

In this environment, also referred to as "static environment," the PDP-11/60 is under control of the UNIX operating system.

2. A dynamic environment where the flight software can be exercised in closed-loop real time. In this environment an airplane simulation is hosted in the PDP-11/60 which runs under the control of the standard RSX-11M DEC operating system.

The following sections will describe the UNIX operating system, the static environment, and the dynamic environment.

5.1 UNIX OPERATING SYSTEM

A standard commercially available version of the UNIX operating system was purchased and modified to make it compatible with the configuration of the verification facility. Features of this system include a CRT terminal (INtext) which provides full-screen text editing, a screen-oriented on-line text editor that runs with the INtext terminal and has multiple "window" capability, a remote job entry (RJE) facility which provides the capability of submitting or receiving jobs from a UNIVAC 1100, a "C" programming language compiler, and an interpretive FORTRAN compiler which accepts limited ANSI FORTRAN 66 code.

The file system consists of a highly uniform set of directories and files arranged in a tree-like hierarchical structure providing:

1. Simple and consistent naming conventions; names can be absolute, or relative to any directory in the file system hierarchy.

2. File linking across directories.

3. Automatic file space allocation and deallocation that is invisible to users.

4. Facilities for creating, accessing, moving, and processing file directories, or sets of these, in a simple, uniform way. Each physical I/O device such as the interactive terminal and the main memory is treated like a file; this results in an easy to use environment where I/O device handlers enjoy the same flexibility and software support as any software file.

5. A complete set of flexible directory and file protection modes which can be set dynamically.

Enhancing the file system is a source-code control system (SCCS) which is a system for controlling changes to files of text (typically, the source code and documentation files of software systems). It provides facilities for storing, updating, and retrieving any version of a file of text, and for recording who made each change, when and where it was made, and why.

SCCS is resource effective, particularly in a development environment which requires the storing of several versions of the source program. In this case SCCS stores only the original version of the program and the subsequent changes made to

it as opposed to all the several versions of the program. If a certain version of the program is needed, the operating system automatically recreates it from the original version and the pertinent changes.

5.2 STATIC ENVIRONMENT

The environment has been designed to meet the requirements of control engineers with only limited interests in system software. The INtext terminal provides a common interface to all the resources required and all the tools can be accessed with simple task-oriented commands from that INtext terminal. The activities supported by this environment include:

1. Creation of a new file or a modification of an existing file.

2. Static verification of a file with or without assertions.

3. Instrumentation of a file for path coverage and the insertion of logical and timing assertions.

4. Error seeding of a file with automated statement selection features.

5. Creation of a CAPS-6 executable load module.

6. Loading of the executable load module in the CAPS-6.

5.3 DYNAMIC ENVIRONMENT

The INtext terminal is the only required operator interface and control station for this environment. From this station different flight cases can be selected, turbulence can be introduced, and the simulation started or halted. The aircraft simulation resides in the PDP-11/60 and transmits flight data in real time to the MDICU. The MDICU converts the data which are then utilized by the flight computers. The flight computers in turn compute control surface commands which are fed back to the flight equations, thus completing the entire loop.

In this mode the PDP-11/60 runs under the DEC RSX-11M version 3.2 operating system appropriately modified for the DFCS environment. The FORTRAN IV Plus compiler and the FP11-E floating point processor complement the system. The DFCS RSX-11M operating system also supports a Printronix line printer, a TS11 magnetic tape unit, an INtext on-line terminal, and a DA11-B interprocessor link (between the PDP-11/60 and the PDP-11/04).

This environment is further enhanced by a direct communications link, between the CAPS-6 and the PDP-11/60, provided by a DR11-C DEC card. This link is utilized by the CAPS-6 to notify the PDP-11/60 of the occurrence of some predefined events, within the CAPS-6 itself, with minimum time delay. This capability is used primarily in support of dynamic testing with logical assertions.

6. FURTHER READINGS

This section contains a comprehensive list of the documents used to describe the DFCS Verification Laboratory. These documents are listed by the section that refers to them. Some contain information used in more than one section and are so noted. Others document the specific configuration of the RDFCS pallet as installed at Ames Research Center rather than a commercial product line. These documents do not have a company document number.

The documents are generally available from the originating company with the exception of proprietary ones. The Rockwell-Collins documents were obtained via Contract NAS2-10270 from the Collins Air Transport Division, Cedar Rapids, Iowa. The documents pertaining to the PDP-11/60 and PDP-11/04 are commercially available from Digital Equipment Corporation, Maynard, Massachusetts. The documents relating to the UNIX operating system are commercially available from Interactive Systems Corporation, Santa Monica, California. The documentation relative to the specific software interfaces within the DFCS Verification Laboratory and the resulting operating environment is available from Hughes Ground Systems Group, Fullerton, California. The documentation relative to the implementation and operation of the software V&V tools is available from General Research Corporation, Santa Barbara, California. The UNIVAC 1100 documents are available from Information Systems Design Division of Control Data Corporation, Santa Clara, California. The HP 2645A Display Station User's Manual is available from Hewlett Packard Corporation, Palo Alto, California.

Following are the documents used as references.

Section	Document
2.1 Digital Flight Control System	1. DFCR-1, Rev. 11, FCS-240 Digital Avionic Flight Control System (AFCS) System Description Document; Rockwell-Collins, Dec. 23, 1980.
	2. DFCR-3, Rev. 3, L1011-500 DAFCS Software Requirements Document; Rockwell-Collins, Aug. 22, 1980.
	3. DFCR-96, Rev. 1, L1011 DAFCS Software Description, 9APRIL80 BASELINE; Rockwell-Collins, June 26, 1980.
	4. Collins Adaptive Processing System (CAPS) Transfer Bus; Rockwell-Collins No. 523-076804-001117, July 15, 1977.
	5. FCC-201 Flight Control Computer Component Maintenance Manual; Rockwell-Collins No. 523-0769387, Jan. 15, 1981.
	6. Introduction to AED Programming, Fourth Edition, SofTech Inc., Dec. 1973.

Section	Document
2.2 CAPS Test Adapter	7. CAPS Test Adapter User's Guide; Rockwell-Collins.
2.3 Modular Digital Interface Control Unit	8. Van Nuys MDICU Monitor Program User's Guide; Rockwell-Collins, Aug. 6, 1979.
	9. CAPS Test Facility Monitor, Version 11.78; Rockwell-Collins.
	10. Van Nuys MDICU Software Summary; Rockwell-Collins No. 790820-STC-MCU-0872, Aug. 20, 1979.
	11. NASA Digital Flight Control System PDP-11/60-MDICU Interface; Rockwell-Collins, Feb. 2, 1981.
	12. NASA RDFCS System Interface Document; Rockwell-Collins, April 8, 1981.
	13. MDICU and Fault Insertion Source Code; Rockwell-Collins.
	14. NASA MDICU Hardware Manual; Rockwell-Collins.
2.4 Servo Simulator	15. NASA Servo Simulator Operators Manual; Rockwell-Collins, April 1, 1981 — see document 12.
2.5 Glareshield Panel	16. GSP-201 Glareshield Panel Component Maintenance Manual; Rockwell-Collins No. 523-0769388-001113, Jan. 15, 1981 — see document 1.
2.6 Breakout Panel	See document 12.
2.7 Discrete Switch Panel	See document 12.
2.8 Buffer Panel	17. 255K-5 Instrumentation Description and User's Manual; Rockwell-Collins, April 20, 1981.
2.9 Other Instruments	18. 331A-8A/8K Horizontal Situation Indicator Overhaul Manual; Rockwell-Collins No. 523-0761581-731113, July 31, 1979.
	19. ADI-55V Attitude Director Indicator Component Maintenance Manual; Rockwell-Collins No. 523-0767197-111113, Aug. 1, 1979.

Section	Document
	20. 327J-5 Mode Annunciator Indicator Overhaul Manual; Rockwell-Collins No. 523-0761934-501113, May 15, 1980.
	21. 914G-3 AFCS Warning Indicator Overhaul Manual; Rockwell-Collins No. 523-0761946-411113, June 15, 1980.
2.10 PDP-11/04	22. Peripherals Handbook; Digital Equipment Corp. EB 18293 20/80 060 09 165.0, 1980.
	23. PDP-11/04/34a/44/60/70 Processor Handbook; Digital Equipment Corp. EB 17716 18/79 090 04 113.4, 1979.
	24. 2645 A Display Station; Hewlett Packard 02645-90001, Jan. 1978.
3.0 Stand Alone Cab	See document 12.
4.0 UNIVAC	25. CAPS Relocatable Cross Assembler User's Guide; Rockwell-Collins No. 108602-01-UG.
	26. Automated Verification of Flight Software User's Manual; Hughes Aircraft Co., May 1982.
	27. Automated Verification of Flight Software User's Manual; General Research Corp. CR-1-974, April 1982.
	28. ISD EXEC-8 User's Guide; Information Systems Design Inc., Nov. 30, 1977.
5.0 PDP-11/60	29. IS/1 User's Guide; Interactive Systems Corp., May 1981.
	30. IS/1 System Manager's Handbook; Interactive Systems Corp., May 1981.
	31. INTERACTIVE System/One Programmer's Manual; Interactive Systems Corp., Oct. 1978.
	32. INTERACTIVE System/One Text Processing Manual; Interactive Systems Corp., May 1980.
	33. RSX-11M Reference Manuals; Digital Equipment Corp.

7. GLOSSARY

ADI	Attitude Direction Indicator
AED	Automated Engineering Design. A high-level programming language for flight software. Similar to Algol, and developed at MIT under USAF sponsorship.
ANSI	American National Standards Institute
ARC	Ames Research Center
ASCII	American Standard Code for Information Interchange
ATS	Automatic Throttle System
AVFS	Automated Verification of Flight Software. An integrated system for the verification of digital flight control software.
A/D	Analog-to-Digital
A/P	Airplane
baud	Bits per second
BCK CRS	BaCK CouRSe LOC
BEAD	An AED bead is a data structure element which can contain an arbitrary number of values of any AED data type; a bead is thus equivalent to a record.
BP	Buffer Panel
CAPS	Collins Adaptive Processing System
CAT III	Category III
CMD, cmd	Command
CPU	Central Processing Unit
CRT	Cathode Ray Tube
CTA	CAPS Test Adapter
CWS	Control Wheel Steering
DEC	Digital Equipment Corporation
DFCS	Digital Flight Control System
DFCSVL	Digital Flight Control System Verification Laboratory
DMA	Direct Memory Access

download	The action of transferring a computer program or routine from a storage device to computer memory through a communications link
D/A	Digital-to-Analog
EIA	Electronic Industry Association
FCC	Flight Control Computer
FD	Flight Director
FORTRAN	FORmula TRANslator
F/B	Foreground/Background
Gelac	Lockheed Georgia Aircraft Corporation
GSP	Glare Shield Panel
GRC	General Research Corporation
HAC	Hughes Aircraft Corporation
HASP	Houston Automatic Spooling Program. A collection of computer programs that provide two-way communications between a front end computer (PDP-11/60) and a main frame computer (UNIVAC 1100) which serves as the host.
HP	Hewlett Packard
Hz	Hertz
IAS	Indicated AirSpeed
ILS	Instrument Landing System
INed	ISC licensed product: a screen oriented on-line text editor. Used with INtext.
INremote	ISC licensed product: special ISC software to support RJE link.
INtext	A CRT terminal sold by ISC: terminal providing full screen multi-window text editing. Used with INed.
ISC	Interactive Systems Corporation
ISD	Information Systems Design
I/O	Input/Output
JCL	Job Control Language. An assembly like language that identifies the input stream to a host system.
K	1024 decimal (from "kilo")
KOPS	K (thousands) of OPerations per Second

LED	Light Emitting Diode
LNAV	Lateral NAVigation
LOC	LOCalizer
LRU	Line Replaceable Unit
LSI	Large Scale Integration
LVDT	Linear Voltage Differential Transformer
MDICU	Modular Digital Interface Control Unit
NASA	National Aeronautics and Space Administration
pif	Pallet Interface Program (Ames developed abbreviation)
PWB	Programmers Work Bench. A version of UNIX, an interactive multi-user operating system developed by Bell Laboratories.
RAM	Random Access Memory
REG	REGister
RJE	Remote Job Entry
RSX-11M	Real-time Resource Sharing Executive, a DEC licensed product: a real-time multi-programming operating system.
SAC	Stand Alone Cab
SAS	Stability Augmentation System
Shell	UNIX terminology, a command interpreter that reads lines typed at the terminal and arranges for their execution.
SPR	Scratch Pad RAM memory
SQL	Software Quality Laboratory, also annotated as SQLab. A set of static and dynamic verification tools for high level programming languages.
S/S	Servo Simulator
TOGA	Take-Off and Go-Around
UNIX	A licensed Operating system: a general purpose, multi-user, time-sharing, interactive operating system. Developed by Bell Laboratories.
VHF	Very High Frequency
VOR	VHF Omnirange Radial
V&V	Validation and Verification
Vs	Computed stall speed

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
NASA TM-84276		

4. Title and Subtitle
An Integrated User-Oriented Laboratory For Verification of Digital Flight Control Systems--Features and Capabilities

5. Report Date
September 1982

6. Performing Organization Code

7. Author(s)
P. de Feo, D. Doane, and J. Saito

8. Performing Organization Report No.
A-9011

9. Performing Organization Name and Address
Ames Research Center
Moffett Field, California

10. Work Unit No.
T-3236Y

11. Contract or Grant No.

12. Sponsoring Agency Name and Address
National Aeronautics and Space Administration
Washington, D.C.

13. Type of Report and Period Covered
Technical Memorandum

14. Sponsoring Agency Code

15. Supplementary Notes
Point of Contact: Ames Research Center
Pio de Feo M/S 210-9
Moffett Field, CA 94035 FTS:448-5048

16. Abstract
A Digital Flight Control Systems Verification Laboratory (DFCSVL) has been established at NASA Ames Research Center. This report describes the major elements of the laboratory, the research activities that can be supported in the area of verification and validation of digital flight-control systems (DFCS), and the operating scenarios within which these activities can be carried out.

The DFCSVL consists of a palletized dual-dual flight-control system linked to a dedicated PDP-11/60 processor. Major software support programs are hosted in a remotely located UNIVAC 1100 accessible from the PDP-11/60 through a modem link. Important features of the DFCSVL include extensive hardware and software fault insertion capabilities, a real-time closed loop environment to exercise the DFCS, an integrated set of software verification tools, and a user-oriented interface to all the resources and capabilities.

17. Key Words (Suggested by Author(s))	18. Distribution Statement
Digital Flight Control System (DFCS) Flight software Verification and validation Real-time simulation	Unlimited Subject Category - 08

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price*
Unclassified	Unclassified	60	A04

*For sale by the National Technical Information Service, Springfield, Virginia 22161